Wikipedia-Based Semantic Enhancements for Information Nugget Retrieval

by

Ian W. MacKinnon

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Mathematics
in
Computer Science

Waterloo, Ontario, Canada, 2008

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Abstract

When the objective of an information retrieval task is to return a nugget rather than a document, query terms that exist in a document often will not be used in the most relevant nugget in the document for the query. In this thesis a new method of query expansion is proposed based on the Wikipedia link structure surrounding the most relevant articles selected either automatically or by human assessors for the query. Evaluated with the Nuggeteer automatic scoring software, which we show to have a high correlation with human assessor scores for the ciQA 2006 topics, an increase in the F-scores is found from the TREC Complex Interactive Question Answering task when integrating this expansion into an already high-performing baseline system. In addition, the method for finding synonyms using Wikipedia is evaluated using more common synonym detection tasks.

Acknowledgements

First and foremost, I'd like to thank my supervisor, Olga Vechtomova, for the guidance and wisdom she has provided in this endeavor. I also appreciate how she gave great deal of latitude in my research topic allowing me to truly take ownership of the following work.

Also, this could not have been accomplished without the help of Charlie Clarke and Gord Cormack for agreeing to read this thesis and for offering direction throughout my entire graduate experience at Waterloo.

I'd really like to thank Mike Patterson of CSCF for going above and beyond when it came to the essential task of systems administration and user support, and Diana Chisholm for ensuring that this work was grammatically coherent.

Lastly, I'd like to thank Christina Boucher, Craig Sloss, Maria Trainer, and Rose Vogt of the Graduate Student Association for making the year I spent as President so smooth in terms of operations that I was able to complete this body of work while still being in office.

Dedication

To my parents, Sue and David, for always being a phonecall away.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

While the classic model of information retrieval revolves around the notion of a document being sought for an information need, the complex interactive Question Answering (ciQA) task introduced at the Text REtrieval Conference (TREC) in 2006[19] focuses on smaller nuggets of text satisfying a user's information need.

This new track more accurately reflects the idea that some users are looking for a piece of information, rather than a document that contains a certain piece of information. While traditional document retrieval systems, such as web search engines, focus on finding a relevant document for a user, the actual piece of information being sought can often be expressed as a sentence or "nugget". This becomes important to the user since additional time is saved not having to search through entire documents to find answers to a question.

The goal of the ciQA task at TREC is to have a system take a query and return a list of relevant nuggets from documents found in the AQUAINT-1 news corpus for that query. However, concepts being sought can have multiple phrases to describe them. It becomes difficult to determine which sentences in AQUAINT-1 news articles contain the query terms being sought, as they may be represented in the parent document by a variety of different phrases still making reference to the query term. For example, if the term "John McCain" was being sought, that specific phrase might appear in an article; however, the sentence which represents the most vital piece of information being sought may simply contain "Senator McCain", an imperfect match.

ciQA topics are meant to model questions about relationships between entities. As we can see from an example ciQA query in Figure 1, there are a number of brack-

eted items we call "facets" which identify the concepts for which the relationship is being sought.

What evidence is there for transport of [drugs] from [Mexico] to [the U.S.]?

Figure 1.1: Example ciQA Query

Many of the ciQA facets are proper nouns and most thesauri, such as WordNet, do not contain entries for these. Thus, a new manner of finding synonyms must be found. In recent years, several new approaches have been proposed to use Wikipedia as a source of lexical information[15, 16, 37], as it can be downloaded in its entirety and contains relatively high quality articles[17].

The anchor description assumption[9] states that the anchor text of a link describes the document it links to. While this holds true for the web in general, it becomes especially useful in the Wikipedia domain since we know that every article effectively describes an entity. As pointed out in previous work about creating an explicit semantic analysis engine based on Wikipedia[16], the anchor text which points to a Wikipedia article contains high quality terms which can be taken as synonyms for the articles which they link to. For example, the article "United States", will have common anchor text such as "U.S.", "America", "American", "United States of America", or "USA".

With over 2 million articles in Wikipedia and 58 million links in between them, there is a large breadth of lexical information available using this approach to synonym-finding. Given the number of links to each article, we also can determine a distribution of how often a particular phrase is used as anchor text to a specific article, and use it to gain insight into how strong a synonym the phrase can be when it is used to label a link to the article.

Our hypothesis is that by mapping facets of the ciQA query to Wikipedia articles, then extracting anchor text to those articles, we can ascertain a list of high-quality synonyms that can be used to improve the retrieval of information nuggets[27].

In order to do this, we will propose an algorithm to automatically select articles which best describe the facets of a ciQA topic in order to extract high quality phrases for expansion.

These expansion phrases will be incorporated into an already high-performing ciQA system. If this method is successful at finding synonyms, we should see an

improvement in ciQA scores. However, ciQA scores can only officially be determined by a human assessor. In order to perform the evaluations more efficiently, an automatic method for evaluating the accuracy of returned nuggets will have to be found.

We will also explore other methods of determining the validity of synonym detection as a method for showing the value of using Wikipedia anchor text for finding synonyms.

The major contributions of this work are:

1. Find the most accurate system for automatic evaluation of ciQA results by comparing system evaluations of ciQA runs with official ciQA 2006 results
2. Conduct a user study to find the most relevant Wikipedia articles for each facet in the ciQA 2006 and ciQA 2007 sets
3. Propose an algorithm to automatically select Wikipedia articles for each facet in the ciQA 2006 and ciQA 2007 sets and show how it returns similar articles as those selected in the user study
4. Show how using the links structure around these articles leads to an improvement in the performance of ciQA systems
5. Show how this method can be used to find synonyms in a general case outside of ciQA

Chapter 2

Literature Review

2.1 Complex Interactive Question Answering

2.1.1 Traditional vs. Complex Question Answering

Question Answering (QA) is a type of information retrieval in which a user gives a query to a system in the form of a question in hopes of getting a specific fact in response. Drawing upon a corpus for data, QA uses more sophisticated Natural Language Processing (NLP) techniques in order to extract the information needed for the user. QA has been a common track at TREC, which has seen thousands of factoid queries be developed as of 2008.

An example QA query for the TREC 2002 track would be:

<top>
<num> Number: 1396
<desc> Description:
What is the name of the volcano that destroyed the ancient city of Pompeii?
</top>

Figure 2.1: Sample QA Topic

The answer is "Vesuvius". This has been a very popular type of information retrieval, but it does not account for a large class of queries for which the answer cannot be expressed in a simple factoid. A prime example of this would be cases where a relationship between entities is the objective.

With the complex interactive Question Answering (ciQA) task introduced at TREC in 2006[19], the focus of evaluation shifted from documents and facts to more elaborate nuggets. In ciQA, templates are used with several bracketed items we call "facets" which are the basis of the information need, as we can see from an example CiQA expansion query in Figure 2.2.

<topic num="27">
<template id="1">
What evidence is there for transport of [drugs] from [Mexico] to [the U.S.]?
</template>
<narrative>
The analyst would like to know of efforts to curtail the transport of drugs from Mexico to the U.S. Specifically, the analyst would like to know of the success of the efforts by local or international authorities.
</narrative>
</topic>

Figure 2.2: Templated Query for ciQA Topic

There are 5 different templates which are employed by in the ciQA task, 2 with 3 facets in the templates, and 3 with 2 facets in the templates. While templated brackets make the task a contrived one, it does allow for the focus of the task to be on the Information Retrieval, rather than entity recognition and Natural Language Processing.

With ciQA, the answer nugget must be a string from an article from the AQUAINT corpora. The ciQA 2006 task uses the AQUAINT-1 corpus, which consists of news articles from the Associated Press, New York Times, and Xinhua News Agency (English version) from 1996 to 2000. ciQA 2007 uses the AQUAINT-2 corpus, which consists of news articles from the Associated Press, New York Times, Xinhua News Agency (English version), Agence France-Presse (English version), China News Agency (English version), and the Los Angeles Times from 2004 to 2006.

Returned nuggets must include the document id from the corresponding article in the corpus to be valid. It is also important that a system use the correct corpus for the test topic year.

2.1.2 ciQA Evaluation

As we can see in Figure 2.3, each topic given in ciQA has a corresponding answer key which gives a nugget which is either a "vital" or "okay" solution to the query. An assessor at NIST will assign a system's responses to the nuggets if they are deemed to be similar. In this sense, the automatic assessment of topics from ciQA becomes problematic.

Topic	Number	Value	Nugget
27	1	vital	Mexico, Switzerland to cooperate on Salinas - Swiss seized over $114 million in bank accounts opened by Salinas
27	2	okay	Anti-drug police in Mexico confiscate 3.5 tons of marijuana
27	3	vital	Mexican heroin trafficking emerges - Mexican authorities discover a new organization smuggling heroin into the US
27	4	okay	Mexican navy seized 20 tons of cocaine off ships traveling Mexico's coast using technology and info supplied by American law enforcement
27	5	okay	Despite the often spectacular seizures and arrests, the bilateral structures to fight drugs put into effect by U.S. and Mexican governments... have been incapable of reducing the intensity of drug trafficking

Figure 2.3: Answer Key for ciQA Topic

The 2006 and 2007 ciQA topics each had 30 topics given to the participants. The success of a system is judged by its F-score, a calculation given by:

$$\mathrm{F}(\beta)=\frac{(\beta^2+1)\cdot\ P\cdot\ \Re}{\beta^2\cdot\ P+\Re} \text{ where,}$$

$$\Re = \frac{r}{R}$$

$$\alpha = 100 \cdot\ (r+a)$$

$$P = \begin{cases} 1 & \text{if } l < \alpha \\ 1 - \frac{l-\alpha}{l} & \text{otherwise} \end{cases}$$

Figure 2.4: Official F-Score Calculation

In the calculations, r is the number of vital nuggets returned by a system, a is the number of okay nuggets, R is the number of vital nuggets in the answer key, and l is the number of non-whitespace characters in the entire system response. The β tuning parameter is given as 3 for the ciQA task.

2.2 Finding Synonymous Terms

One of the fundamental problems of Information Retrieval (IR) is that a given query may have terms referring to the entity being sought, but the precise wording from the query is not used in the text. Synonymy is a problem because many words may have the same or similar meaning. In order to compensate for this problem, IR systems need to be able to find synonyms for query terms.

This problem is also persistent in passage-finding tasks, where a term that exists in a document may be different in the most relevant passage in the document[27]. This can be especially true in news articles where journalists are prone to using multiple terms for the same entity to so as not to seem repetitive in an article.

Here, we explore the different methods of synonym detection in information retrieval.

2.2.1 TOEFL Data Set

Many problems in IR and NLP suffer from the lack of a representative set of data which can be used for comparison of methodologies. This can lead to researchers using their own potentially biased data sets and being unable to make adequate comparisons between systems.

However, synonym recognition does have a well-accepted data set which one can easily use to make comparisons between systems. Dumais and Landauer[22] introduced a trial involving the Test of English as a Foreign Language (TOEFL). In the TOEFL test, a system is given a problem word such as "levied", and four potential synonyms called choice words:

A. imposed
B. believed
C. requested
D. correlated

A system is required to pick the choice word that is the synonym to the problem term. The answers are also given in this test so one can determine if the system returned the correct answer, which in this case was "imposed".

80 TOEFL questions are given in the set which Dumais and Landauer used. The average non-English US college applicant score on the TOEFL dataset was

64.60%[22], giving a minimum standard for systems to strive for in order to be at least as proficient as humans with English as a Second Language (ESL). If a system were to simply guess at the best synonym, a score of 25% could be empirically reached.

However, the TOEFL method of synonym detection evaluation does have a few shortcomings. First, in most real-world situations, a short list of potential synonyms is not given, making the TOEFL problem more of an artificial problem. This makes the problem one of finding the most likely synonym, not retrieving a list of all potential synonyms to a term. Second, the list of terms is rather sparse and does not contain any proper nouns. Languages are productive and a system that could determine synonyms of words in 1998 may not be able to detect synonyms for user queries to an IR system in 2008 given that new words, especially proper nouns, have been developed.

2.2.2 Latent Semantic Analysis

A large thread of research into finding similar words was started by Dumais and Landauer[11] with their Latent Semantic Analysis (LSA). LSA attempts to find the "semantic structure" of terms and documents by looking at their dependencies through a Singular Value Decomposition (SVD) of the term/document matrix.

LSA was originally patented by Landauer et al. in 1988[1] for Bell Research labs and exposed to the research community in 1990[11].

The motivation behind LSA is that while 2 words which mean the same thing may not exist in the same document together, they may co-occur with similar terms in the corpus. Dumais and Landauer were the first to use the TOEFL test as a method of evaluating how well their LSA could perform synonym detection[22].

LSA performs this task by letting X be a m by n term/document matrix, where the element (i, j) is a function of the term frequency of i^{th} term in the j^{th} document. Thus, a row in the matrix is the relation of all documents to a term and a column will be a vector for all the terms in a document.

A SVD is performed on X to give the breakdown:

$$X = U\Sigma\ V^T, \tag{2.1}$$

[1] `http://patft.uspto.gov/netacgi/nph-Parser?patentnumber=4839853`

where U and V are orthonormal matrices and Σ is a diagonal matrix of singular values. Looking at the singular values of Σ, we can choose the top k values by setting the remaining singular values to 0 and choosing the corresponding singular vectors from U and V^T to get the rank k approximation of X, or X_k. X_k becomes a reduced feature-space of X:

$$X_k = U_k \Sigma_k \, V_k^T \tag{2.2}$$

Using this reduced feature-space, each row of U_k is a vector representation of a term which can be compared against another vector to get a similarity score for the co-locates of two terms. $k = 300$ was the value associated with the highest TOEFL score so it is used in all the experiments for LSA on synonym detection.

The Cosine similarity metric is used to determine the similarity between terms. For terms i and j, represented by rows $\hat{t_i}$ and $\hat{t_j}$ from U_k respectively, the similarity would be:

$$\cos\theta = \frac{\hat{\mathbf{t}}_\mathbf{i} \cdot \hat{\mathbf{t}}_\mathbf{j}}{\left\|\hat{\mathbf{t}}_\mathbf{i}\right\| \left\|\hat{\mathbf{t}}_\mathbf{j}\right\|} \tag{2.3}$$

Looking to the TOEFL test, an LSA-based system can take the vector for the given term and the 4 choice words and simply select the choice word with the highest similarity to the problem word to be the synonym. This method yields a TOEFL score of 64.28%, which becomes a reasonable approximation to a score an ESL human could achieve.

There is a problem with LSA in that it relies on a joint Gaussian distribution of words, while a Poisson model for word distribution has been observed[28].

Another limitation of LSA for synonym detection is that it relies heavily on a short list of potential synonyms of which one and only one term is a valid synonym. Otherwise, a term would have to be compared to all the other terms in the corpus to find which have the highest similarity. However, simply because 2 terms are highly related does not necessarily mean that they are synonyms.

2-Stage Approach

While LSA proved itself to be at least as proficient as an average ESL student on the TOEFL test, it was far from the best score which could be obtained. In 2004,

Bhat et al.[3] tried a 2-stage LSA approach to the problem. Their findings were that for a given term, such as 'Al Quaeda' in Table 2.1, the top related terms in a corpus will have a high semantic relation, but few will be true synonyms. In fact, the only synonym given is at rank position 5.

Rank	Term
1	Zubaydah
2	Ressam
3	Raids
4	Hamdi
5	Al Quaida
6	Pakistani
7	Trial
8	Soldier
9	Pakistan
10	Lindh

Table 2.1: Top Terms from LSA for "Al Quaeda" with $k = 20$

The main observation that Bhat et al. make is that if a system knew that "Al Quaeda" was a group entity it would know that synonyms for the term would also be referencing an organization. "Lindh", being a person, could be removed from consideration since it does not fit the ontological pattern. This could be accomplished with the use of an ontology which would know the differences between the entities.

Since such a broad ontology does not exist, the 2-stage LSA system takes the context of each of the top terms found by LSA as a surrogate for an ontology. This is accomplished by first taking the list, L, of the first several hundred terms found by LSA which are at least semantically related. From this, for each term $l \in L$ a document d' is created which contains a text window from around every occurrence of l in the corpus. These artificial documents form the collection D' which is used in the next phase.

LSA is performed a second time on D' in order to get the level of ontological similarity between the terms, which are effectively represented as documents for the second stage. This gives a new list of terms, L' which are believed to have the proper synonyms at a higher ranking in L' than in the original L.

The first test used by Bhat et al. was to get a list of synonym pairings of chemical compounds. In Table 2.2, we can see the terms used and their ranks at

Name	Target Synonym	Stage 1 Rank	Stage 2 Rank
Methanal	Formaldehyde	19	2
Ethanal	Acetaldehyde	22	13
Propanal	Propionaldehyde	100	7
Butanal	Butyraldehyde		10
Propanone	Acetone	53	15
Ethene	Ethylene	67	11
Propene	Propylene	20	14
Ethenyl	Vinyl		76
Propyne	Acetylene		47
Methanol	Methyl alcohol	150	89.5
Ethanol	Ethyl alcohol	113.5	58
2-Butanone	Ethyl methyl ketone		12.3
2-Propenyl	Allyl		45
Aminobenzene	Aniline	75	4
Hydroxybenzene	Phenol	53	1
Phenylmethanal	Benzaldehyde	25	3
Pentanal	Veraldehyde		11
Dichloromethane	Methylene chloride	274	16
Nonanal	Nonylaldehyde		14
Pentanedial	Glutaraldehyde	9	15
Cyclohexene	Tetrahydrobenzene	127.5	
Methylpropene	Isobutene	122	1
Bromocyclohexane	Cyclohexyl bromide	13	
Nitromethane	Nitrocarbol		21

Table 2.2: Rank Results for Chemical Compounds test

the stages of the given algorithm. The Stage 1 ranks are basic LSA and Stage 2 ranks incorporate the ontological similarity measure.

The "average rank" of the terms was used to show where the synonyms existed on average in the top 400 terms given by the respective stages of the method. The average rank of the synonym in the Stage 1 was 82, whereas the average rank of the synonym after Stage 2 was 28. Thus, the proper synonym was ranked much higher using the ontological similarity of the second run of LSA.

To further demonstrate the validity of the method, the TOEFL test was used. Similar to basic LSA, the highest ranked of the 4 choice words on the similarity list to the problem word was chosen as the synonym. This led to a score of 74.4% on the 80 questions, a reasonable improvement on the 64.28% of the baseline LSA.

While the 2-stage LSA does give a reasonable improvement over traditional

LSA, it makes a large assumption by stating that the context window around a term can be used to find its position in an ontology. The example given is that an organization should not be considered as a synonym for an individual. However, this touches on one of the most basic examples of metonymy. ORGANIZATION-for-PERSON is a type of metonym in which actions performed by a person are attributed to organization for which they are a part[29]. An example is: "The United States said today that it would not ratify the Kyoto treaty." Clearly, the United States is a political entity and cannot speak, but the President or other government official could have spoken on its behalf. Thus, the context window around a term is not the most reliable method of determining ontological location.

2.2.3 Point-wise Mutual Information

Pointwise Mutual Information (PMI) was introduced by Church et al. as an objective measure of word similarity based on word co-occurence[8].

PMI based on data obtained from Information Retrieval (PMI-IR) is based on a similar notion to LSA in that "a word is characterized by the company it keeps"[14]. Co-occurrence of terms becomes the fundamental facet in determining similarity.

In the next sections we will explore how this can be used to determine how likely it is that two words are synonymous.

Initial PMI Work on Synonyms

Turney has made 2 large contributions to this particular field. The first was his 2001 work[40] in which he made a comparison of PMI against LSA.

In order determine the similarity of two terms, k being the problem word, and c_i being i^{th} choice word, the PMI score similarity is based on:

$$score(c_i) = \log\left(\frac{p(k \cap c_i)}{p(k)p(c_i)}\right) \tag{2.4}$$

This is simply a function of whether the terms are more likely to co-occur than to occur independently. The logarithm of this simple independence test gives us the amount of information we can ascertain from the likelihood of terms co-occurring.

Turney uses 3 different implementation functions for this similarity measure using the AltaVista search engine. The function $hits(query)$ is used to give the number of documents found by AltaVista for a given $query$.

The first equation is the most literal interpretation of PMI, which simply counts the number of times the words occur together over the number of times the problem word appears:

$$score_1(c_i) = \frac{hits(c_i\ AND\ a)}{hits(c_i)} \tag{2.5}$$

The second equation makes use of the AltaVista "NEAR" operator, which only takes into account documents where the two terms in the given query are close to one another. Similar to the first equation, the second counts documents in which the two terms occur, but in this case, they must occur within 10 words of each other to be counted as a co-occurrence:

$$score_2(c_i) = \frac{hits(c_i\ NEAR\ a)}{hits(c_i)} \tag{2.6}$$

The third and final equation tested by Turney also makes use of the "NEAR" operator, but in this case ensures that the two co-occurring terms do not have a "not" nearby, in an attempt to account for negation in the discourse.

$$score_3(c_i) = \frac{hits((c_i\ NEAR\ a)\ AND\ NOT\ ((a\ OR\ c_i)\ NEAR\ "not"))}{hits(c_i\ AND\ NOT\ (c_i\ NEAR\ "not"))} \tag{2.7}$$

In order to use the TOEFL evaluation, Turney's system would calculate the $score_j$ for $j = 1, 2, 3$ between the problem word and each of the choice terms. Whichever choice word had the highest score was considered to be the synonym. Using this method, Turney obtained the TOEFL scores in Table 2.3

Method	$score_1$	$score_2$	$score_3$
Score	62.5%	72.5%	73.75%

Table 2.3: TOEFL Test Scores using Turney's PMI-IR Functions

As we can see, the best TOEFL test was obtained using the $score_3$ function. PMI-IR methods seems to give performance at par with the best LSA methods;

however, PMI-IR has a major advantage in that it is a simpler implementation since the computational requirements to perform a SVD of the term-document matrix X for LSA are quite expensive. The 2-stage method would clearly take an even longer amount of time given the required second stage LSA on top of the initial LSA.

PMI-IR on the other hand only requires an initial indexing of most IR systems in addition to looking at postings lists for terms in order to perform the binary operations.

Co-locate Windows

PMI-IR shows a solid approach to the synonym detection problem; however, the "NEAR" operator given by the AltaVista search engine is hard coded to look for 10 terms on either side of the given word. An interesting problem arises in regards to what constitutes a "co-occurrence" of two words.

In 2003, Terra and Clarke conducted a study to determine what constitutes a "co-occurrence" of terms, be it document, phrase or window[38]. A corpus of Web data was gathered from a general crawl of the web in 2001, yielding a terabyte of documents for experiments to be performed on.

Terra and Clarke looked at PMI, χ^2, Likelihood ratios, and Average Mutual Information, as well as their respective performance with window- and document-oriented approaches for word co-occurrence. Window-oriented approaches considered two terms to co-occur if the two terms were within N terms of each other. In this case N is the window size and different sizes were evaluated. In document oriented approaches, two terms were said to co-occur if they both occured in the same document.

It was found that PMI-IR methods both performed very well compared to the other methods when evaluated on the TOEFL task. However, document-based PMI-IR had the best performance with 81.25% correctness on the TOEFL test. When using a window approach, a window of 16-32 terms was found to be the optimal size.

The results were slightly better than the Turney method of document PMI-IR which yielded 73.75%. This was explained by the fact that Terra and Clarke were using their own crawl of the web, whereas Turney was using AltaVista, which used a proprietary collection system for Web data.

Tweaking Corpus Size and Comparison

A 2006 effort by Bullinaria and Levy experimented with different methods of comparison of terms with PMI-IR[6] on the TOEFL test. Specifically, they experimented with only allowing positive PMI-IR values, as well as using Cosine similarity in the PMI-IR calculation.

Given that PMI-IR was the best performing mechanism, Bullinaria and Levy focused their attention on different vector types, comparison operators, and corpus sizes. They found that using the Positive PMI Cosine score yielded the best results on the TOEFL test.

Interestingly, when varying the window length for what is deemed a co-occurrence, Bullinaria and Levy found that a smaller window of 4 or 5 words performed the best with the Positive PMI Cosine method.

In terms of corpus size, it was found that using the British National Corpus (BNC) with the Positive PMI Cosine scoring method, a TOEFL score of 85% could be reached. This is a considerable improvement on Turney's Initial PMI-IR work.

Bullinaria and Levy found that the larger the corpus, the better the performance. They came to this conclusion after restricting the size of the BNC and found weaker performance. It was admitted that the BNC was a significantly larger 90 million words compared to the 4.6 million word Grolier's Academic Encyclopedia corpus used by Dumais and Landauer[11] for their LSA experiments. Thus, it is hard to draw hard conclusions about LSA versus PMI-IR when using different corpora.

Another attribute which Bullinaria and Levy gave credit for their high score was the quality of the articles in the corpus. When comparing to newsgroup postings, blocking for number of words in the corpus, there was on average a 10% improvement by using the BNC. However, the size advantage BNC had could be compensated for by adding more newsgroup postings to give more statistical information for the system.

While Bullinaria and Levy gained an impressive 85% score on the TOEFL test, most of their contribution can be attributed to testing for larger corpora size, which was the main factor in their improvements. It therefore becomes clear that having a large, high-quality corpus is necessary for statistical measures of word similarity.

2.2.4 Thesaurus and WordNet

So far, we have looked at two corpus-based statistical methods of determining synonyms, LSA and PMI-IR. However, specific tools already exists for the lookup of synonyms which should be fairly easy to adapt for our purposes. The first method will look at an adapted WordNet-based method, and the second is a Roget's thesaurus-based method.

Resnik's Semantic Similarity

A large body of work has been produced on finding how much semantic similarity exists between two terms. One such method is Resnik's WordNet approach[34] introduced in 1995. In this method, Resnik counted the WordNet "IS-A" distance between 2 words. His initial goal was to approximate the similarity scores of 30 pairs of nouns in the Miller and Charles set[31], which gives a human-assessed score of 0.0 to 4.0 for similarity of the terms.

In this task, Resnik scored a correlation of 0.79 to the human judges using his method of measuring how much information two concepts have in common. Formally, the similarity between two concepts c_1 and c_2 is:

$$sim(c_1, c_2) = max_{c \in\ S(c_1,c_2)}[-\log(p(c))], \tag{2.8}$$

where $S(c_1, c_2)$ is the set of concepts which subsume both c_1 and c_2. In the case where multiple senses or concepts exist for a term, the semantic similarity is judged to be the highest gained from the most similar sense.

The 0.79 correlation score is fairly high given that any individual human assessor can only get a correlation of 0.9. 0.9 was found to be the correlation score of the 38 undergraduate students who helped create the Miller and Charles set. Jarmasz and Szpakowicz attempted to translate these high semantic similarity scores to the TOEFL set[18].

When Resnik's method was applied to the TOEFL set by selecting the choice word with the highest similarity to the problem word, a score of only 21.3% was obtained: less than if one were to simply select one of the potential synonyms at random.

Jarmasz and Szpakowicz gave the following explanation for the dismal performance: "Most of the WordNet-based systems perform poorly at the task of answer-

Length	Description	Example
0	the same semicolon group	journey's end - terminus
2	the same paragraph	devotion - abnormal affection
4	the same part of speech	popular misconception - glaring error
6	the same head	individual - lonely
8	the same head group	finance - apply for a loan
10	the same sub-section	life expectancy - herbalize
12	the same section	Creirwy(love) - inspired
14	the same class	translucid - blind eye
16	in the same Thesaurus	nag - greased lightning

Table 2.4: Path Lengths in Roget's Thesaurus

ing synonym questions. This is due in part to the fact that the similarity measures can only by calculated between nouns, because they rely on the hierarchical structure that is almost only present for nouns in WordNet. The systems also suffer from not being able to deal with many phrases."[18].

Clearly, the coverage of WordNet becomes a liability for such a measure. But it also illustrates a larger problem with such "look-up" systems, which is that if a word does not exist in the man-made table, it is unable to find any related terms at all.

Roget's Thesaurus

In addition to providing a comparison of other semantic similarity measures applied to the synonym detection task, Jarmasz and Szpakowicz also introduced their own synonym detection method based on the 1987 version of the Roget's Thesaurus[18].

The method proposed by Jarmasz and Szpakowicz is meant to overcome a perceived shortcoming in Resnik's "IS-A" counting method: not all WordNet relations should be considered equal. Their solution was to introduce multiple distances, shown in Table 2.4, for which the length between two terms is given as the shortest distance between them.

The formal function used to determine the measure of semantic similarity between two words w_1 and w_2 is:

$$sim(w_1, w_2) = 16 - [\min distance(r_1, r_2)], \tag{2.9}$$

where r_1 and r_2 are sets of reference for w_1 and w_2, respectively.

While the Resnik method garnered a correlation of 0.79 on the Miller and Charles noun pairs, the Jarmasz and Szpakowicz method based on Roget's achieved a correlation of 0.88. When looking at the TOEFL set, if one were to choose the choice word with the highest Roget's similarity score to the problem word, a score of 78.75% is achieved.

Jarmasz and Szpakowicz make the claim that: "We have shown in this paper that the electronic version of the 1987 Penguin Roget's Thesaurus is as good as, if not better than, WordNet for measuring semantic similarity."[18]

While the results are indeed significant, it is presumptuous to state that Roget's Thesaurus is better for this task than WordNet. First, the Resnik method only took into account a small aspect of WordNet relations, whereas the Roget's model had a more fine-grained method for assigning scores which took into account a large amount of the information available in the Thesaurus. For Roget's to really be considered a better lexical source than WordNet, one would have to apply the same fine-grained semantic similarity function to WordNet that was given to Roget's and still get similar results. Otherwise the comparison is not really fair.

The second issue with the Roget's method is that it can easily suffer the from the same problem which led to the dismal performance of WordNet on the TOEFL set: missing terms. While it is clear that Roget's Thesaurus has better coverage of the English language than WordNet, it still does not cover every eventuality. While TOEFL does not contain proper nouns or nouns new to the English lexicon, it is perfectly plausible that such terms could require synonym detection in a normal IR task.

With LSA and PMI-IR models, it is at least possible for the newer terms to exist in the base corpus to allow for comparison.

2.2.5 Compound Methods

In 2003, Turney made his second contribution to the synonym detection field when he proposed a combination approach in to the task in which multiple modules, each representing a previous method, were combined to make a decision for the choice word[39].

Turney et al.'s goal was to combine the outputs of TOEFL selecting modules for LSA, PMI-IR, Thesaurus, and Connector. LSA was performed by directly querying

the web interface for LSA[2]; PMI-IR used the $score_3$ method from Turnery's 2001 work[40]; Thesaurus was accomplished using the Wordsmyth[3] online thesaurus and looking for overlap for choice words; Connector by querying Google[4] for the pairs of words in the snippets on the results page.

With these four modules, it was necessary to join the results together to make a final decision in regards to a choice word. Turney experimented with a mixture of rules, which simply assigned weights to all the modules to get a harmonized score; the logarithmic rule, which combined the logarithms of the output; and the product rule, which multiplies the outputs. The product rule was found to have the best results, for which the probability of selecting choice j is:

$$D_j^{h,w} = \frac{P_j^{h,w}}{\Sigma_j P_j^{h,w}}, \tag{2.10}$$

where

$$P_j^{h,w} = \Pi_i (w_i p_{ij}^h + \frac{(1 - w_i)}{k}), \tag{2.11}$$

where w_i is the weight assigned to module i's output for an instance h.

In order to find the appropriate weights, w_i, Turney created a synonym set of 431 synonym problems from crossword puzzles, TOEFL questions, other ESL questions, and Reader's Digest puzzles. These were then split into a training set of 331 problems and a test set of 100 problems. After training, the system was run on the 100 test synonyms and garnered a score of 80%. However, when run on the TOEFL set, the system achieved an astounding 97.5% score.

The other interesting aspect of this work is that none of the individual modules were in their prime tweaked forms, as presented in earlier sections. In fact, all the modules were accessed by Turney through web interfaces, making the experiments relatively easy to conduct. It would be interesting to see the effects of all the individual modules being properly tuned beforehand.

2.2.6 Results

Looking at Table 2.5 we can see a summary of all the different methods applied to the TOEFL test set. Given the 97.5% score by Turney's combination work, the

[2] `http://lsa.colorado.edu`

[3] `http://www.wordsmyth.net`

[4] `http://www.google.com`

Method	Description	TOEFL Score
WordNet Distance	Resnik's Word Similarity score applied to TOEFL	21.3%
Guessing	Random Choice for Synonym	25%
LSA	Dumais and Landauer's Initial LSA	64.28%
ESL Student	Average Non-US TOEFL Score	64.6%
PMI-IR	Turney's Initial PMI work using AltaVista and it's "NEAR" operator	73.75%
2-Stage LSA	Bhat et al's use of 2 LSA steps for Ontological similarity	74.4%
Thesaurus Distance	Jarmasz and Szpakowicz's similarity based on Roget's thesaurus	78.75%
PMI-IR	Terra and Clarke's tweaking of PMI-IR window sizes and methods on own web crawl	81.25%
PMI-IR	Bullinaria and Levy using the BNC and the Positive PMI Cosine comparison	85%
Compound	Tuney et. al's compound module method with simple learning	97.5%

Table 2.5: Summary of TOEFL Scores

TOEFL data appears to be a solved problem. However, given that on his own test set the system which garnered the top TOEFL score only achieved an 80% success rate, synonym detection is far from being a solved problem itself.

As stated earlier, there is an issue in that picking a single synonym which is known to exist from a short list of choice words may not be the most appropriate evaluation of synonym detection. While an open-ended synonym search would be more useful in an IR sense, it would also require a more elaborate evaluation mechanism taking into account precision and recall.

Such a data set does not exist, but given that the TOEFL test has seen remarkable improvement, it is likely the next logical step in the problem of synonym detection. It may also be necessary in order to push the field into a direction such that it becomes practical to have an entire thesaurus built automatically.

2.3 Wikipedia as a Source for Lexical Information

Wikipedia was launched in 2001 as a "free encyclopedia that anyone can edit" by Jimmy Wales and Larry Sanger. As of January 1, 2008, the encyclopedia contains over 2.1 million articles in the English version and is available in over 200 languages[5]. The novel feature of Wikipedia is the ability for any user to modify the articles within it. While this has led to many criticisms of the accuracy of Wikipedia given the potential for vandalism, biased articles, and a valuing of consensus over accuracy, a recent study in Nature found that the average number of inaccuracies in Wikipedia science articles averaged 4 per article, whereas Encyclopedia Brittanica averaged 3[17], making the two sets roughly equivalent in accuracy.

What makes Wikipedia useful to the NLP and IR communities is that the entire corpus can be downloaded in its XML format so researchers are able to make use of it without having to crawl the whole site[6]. These XML outputs are made available every month, reflecting the most recent changes to the encyclopedia.

A few years after Wikipedia's launch, there were several academic projects to either improve Wikipedia[1], study its structure[42], or ascertain its quality[33]. While there were some attempts to use the information in the collection as a data source for traditional Question Answering[2], the first use of Wikipedia as a source for lexical information was by Bunescu and Pasca[7] in 2006 to disambiguate named entities.

2.3.1 WikiRelate

Strube and Ponzetto used Wikipedia as part of their 2006 WikiRelate system[37] in order to determine the semantic relatedness of two terms. In this work, it was noted that WordNet does not contain information about named entities such as "Condoleezza Rice" or "the Rolling Stones", nor about specialized concepts such as "excytosis". By comparison, Wikipeida has a broader coverage and is updated at a much more rapid pace.

Wikipedia has also incorporated a form of taxonomy by allowing users to group articles into categories. While this is not a very well-structured taxonomy of con-

[5] `http://wikipedia.org`
[6] `http://download.wikipedia.org`

Dataset	WordNet correlation coefficient	Wikipedia correlation coefficient
Miller & Charles	0.82	0.41
Rubenstein & Goodenough	0.86	0.5
353-TC	0.34	0.48

Table 2.6: Leacock and Chodorow Similarity Correlation on WordNet and Wikipedia

cepts in the encyclopedia, it is an example of a "folksonomy". A folksonomy being a collaborative approximation to a proper ontology for the encyclopedia. WikiRelate attempts to use the category information from Wikipedia in order to determine the semantic distance between a pair of terms.

Experimenting with edge-based disatance measures, Strube and Ponzetto found that the path-length measure proposed by Leacock and Chodorow[24], originally designed for counting WordNet edges, gave the best results for measuring sematic similarity, be it on WordNet or Wikipedia categories. The Leacock and Chodorow method for determining the similarity between concepts c_1 and c_2 is formally given as:

$$lch(c_1, c_2) = -\log\left(\frac{length(c_1, c_2)}{2D}\right), \tag{2.12}$$

where $length(c_1, c_2)$ is the number of nodes along the shortest path between the concepts and D is the maximum depth of the taxonomy.

In order to test the use of WordNet versus Wikipeda for sematic relatedness, Strube and Ponzetto used three standard datasets, namely the Miller & Charles list of 30 noun pairs described earlier[31], the 65 word pair superset of Miller & Charles, the Rubenstein & Goodenough list[36], and the WordSimilarity-353 collection[13], which is a superset of both.

Using the datasets and the Leacock and Chodorow similarity score with both Wikipedia and WordNet, Strube and Ponzetto observed the results in Table 2.6.

Strube and Ponzetto observed that the reason Wikipedia performed better than WordNet on a large dataset compared to the smaller ones had to do with sense proliferation. With no sense disambiguation in the Wikipedia set, the shortest paths between the words would often be found using senses from the set which did not represent the true meaning.

The major contribution of Strube and Ponzetto was showing that the Wikipedia

category system can be used as an effective source of lexical information. However, this was only true of larger datasets given problems with many word senses in Wikipedia. While this shows that Wikipedia can be used by the NLP community, it does not make full use of the Wikipedia articles or its link structure; only its category information. In our next section, we will look at the contributions of Gabrilovich and Markovitch as they build a semantic interpreter to better use Wikipedia to determine semantic similarity.

2.3.2 Explicit Semantic Analysis

In 2007, Gabrilovich and Markovitch [16] used Wikipedia combined with a vector space model in order to produce their Explicit Semantic Analysis (ESA) system. ESA uses Wikipedia in order to create a high-dimension space of concepts by using the articles of Wikipedia as a feature space.

The first step in ESA is to transform a given text fragment into a Term Frequency/Inverse Document Frequency (TFIDF) attribute vector. For each word in the text, $w_i \in T$, t_i becomes the TFIDF value for w_i in the text fragment. Letting $< k_j >$ be an inverted index of entries for words, k_j becomes the strength of association of word w_i with concept (described by a corresponding Wikipedia article) $c_j \in c_0 \cdots c_n$, where n is the total number of Wikipedia articles/concepts in the collection. In order to obtain the semantic vector, V, we take the j^{th} entry in the vector of length n to be:

$$< v_j >= \sum_{w_i \in T} t_i \cdot k_j \tag{2.13}$$

Essentially, each element of the semantic vector becomes a summation over all words in the text of the weight of the word in the text fragment multiplied by its relation to the concept article for that entry. Once two sets of text are represented as semantic vectors, they can be compared using the cosine similarity metric. This process is shown in Figure 2.5.

In order to reduce the feature space and ensure that only significant concepts were considered in the analysis, Wikipedia articles that were fewer than 100 words in length or had fewer than 5 incoming or outgoing links were ignored. This led to 241,393 articles left for use in the semantic feature space. While this a significant pruning of Wikipedia articles, it is still contains more entries than WordNet.

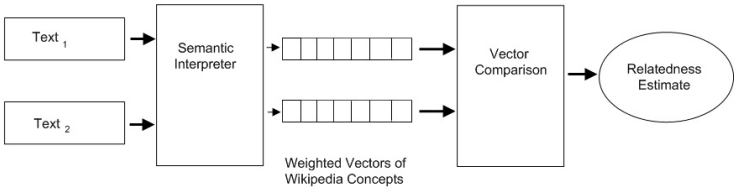

Figure 2.5: ESA Semantic Interpreter

Method	Correlation with Human Judges
WordNet [18]	0.35
Roget's Thesaurus[18]	0.55
LSA[13]	0.56
WikiRelate[37]	0.48
ESA[16]	0.72

Table 2.7: WordSimilarity-353 Score Comparison

This type of semantic analysis is explicit, given that the fragment of text is related to semantic concepts given by Wikipedia directly, rather than by the latent concepts which Latent Semantic Analysis resolves to using a SVD. Once the semantic vectors have been made for text they can be compared for relatedness by simply using Cosine similarity measures.

In order to determine the validity of this method of determining semantic relatedness, the WordSimilarity-353 Collection[13] previously described was used. Each pair in the set is compared by converting each of the words into the semantic vectors and performing a comparison by cosine similarity to get a relatedness score.

Using this method for calculating semantic relatedness, we can see the Pearson's correlation co-efficient scores on the WordSimilarity-353 to human judges in Table 2.7.

ESA with Wikipedia may be a simple vector-space method but using Wikipedia articles explicitly as concepts becomes a very useful tool that surpases the WordNet inspired methods given by WikiRelate. Gabrilovich and Markovitch note explanations for the ESA/Wikirelate differences. Firstly, ESA does not require that the words being looked up are titles in Wikipedia, only that they exist in the collection. Second, ESA can handle whole fragments of text and not just individual words. Lastly, WikiRelate only looks at category information for its best performing method where ESA takes into account the whole text of Wikipedia.

2.4 Anchor Text to Improve Information Retrieval

In the task of web retrieval, the link structure surrounding a document has been used as a mechanism to assist in IR. Methods such as PageRank[32] and HITS[20] use the graph formed by hyperlinks to gain addition insight into the potential relevance of a document on the web.

However, recent work has begun using anchor text alone as a method of improving retrieval. One of the first uses of this approach was done in 2001 by Craswell et al[9].

In this work, Craswell et al attempted to replace a document with a surrogate which consisted entirely of text from the anchor text of the links that referenced that document. This is based on the anchor description assumption that the anchor text of a link describes the document it links to. More frequently occuring anchor text would occur more frequently in the surrogate document, for example:

> If 7332 pages link to `http://www.excite.com` with the anchor text "excite", that word is added 7332 times to the anchor document.[9]

They contrast using the anchor documents versus standard content while using BM25 retrieval on a collection of TREC documents with the objective of finding specific websites.

100 randomly chosen sites from the VLC2 collection were chosen for the test set. In the experiments, user satisfaction at the first result (P@1) was found to be 15% for original content, and 35% for anchor documents. Thus, this method shows that anchor text can become extremely useful for navigation-type queries on web documents given the strong advantage found using anchor documents. However, navigation-queries are only one type of information need on the web.

In the next Chapter we will see how the previous work in synonym detection and anchor text can be used as a foundation to improve an already high-performing ciQA system.

Chapter 3

System Foundations

3.1 Baseline System

Before we can begin to integrate Wikipedia-based enhancements into the ciQA realm, we must first find a suitable ciQA system which we can modify. We base our system on the one which yielded the highest F-scores for initial automatic runs, which we refer to as the Giraffe system[41]. The system described will be the one which generated the **UWATCIQA1** run at the TREC 2006 ciQA task for the Question Answering track.

At a high level, the system works by getting a list of documents from the collection, parses sentences, and returns a ranked list of sentences as output for evaluation.

To gain an initial set of documents, the system parses out the initial topic to get the 2 or 3 facets from the test topics, performs a document retrieval using the facet words as query terms, and returns the top 200 documents from the AQUAINT newswire corpus. Once a list of documents has been retrieved, every document is split into candidate sentences. Afterwards, a score of 0,1,2, or 3 is assigned to a sentence depending on the number of facets which are represented in a candidate sentence. The top 30 sentences are returned as the response, removing similar sentences to ensure all returned sentences are novel.

3.1.1 BM25

In order to obtain an initial set of documents, we are first required to have a mechanism to rank documents in the AQUAINT corpora according to their relevance for a query. For the creation of this ciQA system, we are using Okapi BM25 ranking[35].

BM25 takes a query, Q, which has keywords $q_1, q_2..., q_n$, and a document D, and assigned a BM25 score to the document based on the formula:

$$score(D,Q) = \sum_{i=1}^{n} IDF(q_i) \cdot \frac{tf(q_i,D) \cdot (k_1+1)}{tf(q_i,D) + k_1 \cdot (1 - b + b \cdot \frac{|D|}{adl})}, \tag{3.1}$$

where $tf(q_i, D)$ is the term frequency of query term q_i in document D, $|D|$ is the number of words contained in the document, adl is the average document length, k_1 and b are free parameters where normally $k_1 = 1.2$ and $b = 0.75$.

One of the major aspects of BM25 is that the words from a query found in a document are weighted by their Inverse Document Frequency (IDF) score in the absence of relevance judgements. This ensures that scores are not biased towards documents that simply contain a large number of common words. The IDF of a term is given as a function of the number of documents in a collection over the number of documents which contain at least one instance of a term. Thus, sparsely occurring terms have a higher information score paired with them and frequently occurring ones have a much lower score.

More formally, the IDF weight of a term q_i is computed as:

$$IDF(q_i) = \log \frac{N - n(q_i) + 0.5}{n(q_i) + 0.5}, \tag{3.2}$$

where N is the total number of documents in the collection and $n(q_i)$ is the number of documents which contain at least one instance of the term q_i.

While BM25 is just one of many document ranking mechanisms, it was found that the information retrieval mechanism did not have a significant impact on question answering compared to NLP techniques used to extract information from the top retrieved documents[26].

3.1.2 ciQA 2006 System

We will eventually be making modifications to the Giraffe system[41], but first we will describe the system and how it performed the ciQA retrieval.

First, for each topic a query is build and the top n documents are retrieved using the Okapi retrieval system The query is built as follows:

For each topic in the ciQA set, the top 200 documents retrieved using a BM25 ranking are collected. The query input into the BM25 function is created by taking all of the facets from the topic template save for the first "relationship" facet. For example, if the template were "What [relationship] exist between [entity] and [entity]?", the first facet "relationship" would not be joined with the others to form the BM25 query. The only exception is if the relationship facet were "financial ties", in which case the words "financial, money, funds, monetary" are added into the query. Also, all proper nouns identified in the narrative section of the topic are identified using Brill's Part-of-Speech tagger[4] and concatenated into the query as well.

Once the 200 documents have been retrieved, the existing facets are expanded by adding pertainyms identified with WordNet. For example, the adjective "American" is expanded to include its pertainym "America".

The next step is to determine which documents are "valid". A valid document contains at least one proper noun from each facet if any facet contained a proper noun. If none of the facets contain proper nouns, then the whole document set is considered valid. Invalid documents are dropped from consideration.

All valid documents are split into sentences and ranked by three things:
1. Number of facets the sentence contains. A sentence is considered to have a facet if at least one word from that facet is present. Discard all sentences that have no facets.
2. Resolve ties by the number of query terms the sentence contains.
3. Resolve ties by the number of lexical bonds the sentence has with the following sentences in the document. A sentence is said to have a lexical bond with another sentence if they have at least one lexeme in common.

The top 30 sentences are returned by the system for each ciQA topic.

3.1.3 2006 Performance

As we can see in the 2006 ciQA results[19] in Table 3.1, the **UWATCIQA1** run based on the Giraffe system described above had the highest performance of all the automatic runs submitted to NIST for ciQA.

Organization	Run Tag	F-Score
U. Mass.	UMASSauto1	0.133
CL Research	clr06ci1	0.151
U. Mass.	UMASSauto2	0.171
CL Research	clr06ci2	0.175
MIT	csail1	0.203
U. Maryland	UMDA1pre	0.224
U. Waterloo	**UWATCIQA1**	**0.247**

Table 3.1: Initial Results of Automatic Systems from ciQA2006

Other runs such as manual and interaction were not included since they both had some degree of human interaction to determine their outputs. Given the large margin which the **UWATCIQA1** f-score leads by, it becomes appropriate to base our Wikipedia semantic enhancements on this system, given it is already a high baseline.

3.1.4 System Description

The baseline system we will use in our experiments will differ slightly from the 2006 Giraffe system. More specifically, the base system performs as follows:

1. Parse out the initial topic to get the 2 or 3 facets from the test topic. In the case of the topic given in Fig 3.1, this would be "Hank Aaron" and "Barry Bond's use of steroids".

```
<topic num="55">
<template id="4">
What is the position of [Hank Aaron] in regards to [Barry Bond's
use of steroids]?
</template>
</topic>
```

Figure 3.1: Templated Query for ciQA 2007 topic

2. Split apart each of facets into single terms, join all the terms for a topic together into one list and perform a BM25 retrieval[1]. The top 200 documents from the AQUAINT newswire corpus are returned by the system and kept in order according to their BM25 scores.

3. Since we are interested in nuggets which contain information about the relationship between all of the facets, we need to ensure that all of the concepts noted in the test topic are represented in the document somewhere. For each of the returned documents, we wish to determine the validity of the document by ensuring at least one non-stopword from each of the facets exists in the document. From our Figure 3.1 example, a document is considered valid if and only if the text of a retrieved document contains a non-stopword term from the facet "Hank Aaron" and "Barry Bond's use of steroids". Invalid documents are moved to the end of the list of 50 documents, effectively giving us an ordered list of valid documents sorted by BM25 followed by invalid documents sorted by BM25.

   We can see a demonstration of this in Fig 3.2.

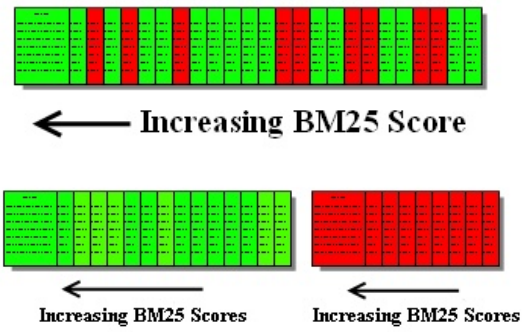

Figure 3.2: Document ordering before and after validity sorting.

4. At this point we have a list of documents which likely contain information about the test topic; however, our goal is not to retrieve a list of relevant documents like many other TREC tasks, but rather to return a list of nuggets. To do this we break up the documents from the top 200 into their constituent sentences. In order to preserve the rankings provided to us by BM25, we keep the sentences in order in which their parent document occurred in the top 200 ranking, see Figure 3.3.

[1]Using default parameters k=1.2, b=0.75

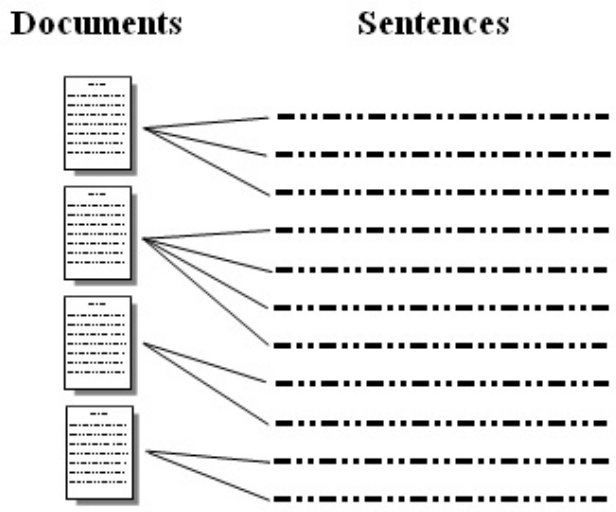

Figure 3.3: Splitting Documents into their Constituent Sentences

5. We next require a method of ranking sentences for return. To do this, we will apply a score of 0,1,2, or 3 to a sentence depending on the number of facets which are represented in a candidate sentence. Each topic will have 2 or 3 facets containing a number of terms within them. For each facet, let us consider $\Gamma = \{\gamma_1 ... \gamma_n\}$ to be the set of non-stopword terms for a facet in a ciQA topic.

   A score is assigned to a candidate sentence S, by iterating through all the γ_i in Γ, and determining if any of the non-stopword stems of the terms exist in the sentence. If at least one exists, a nugget is said to be represented in the facet. More formally:

$$score(S, \Gamma) = \begin{cases} 1 & \text{if at least one of } \gamma_i \in \Gamma \text{ exist in S} \\ 0 & \text{otherwise} \end{cases} \tag{3.3}$$

   We get the total score for S by taking the sum of the $score(S, \Gamma)$ for each facet in the topic.

   A sort is then performed on the list of sentences, but it is of great importance that the sort preserve the original ordering of the sentences with the same score. This allows for sentences which come from a document with a higher BM25 score to be ranked higher given that they are likely more relevant to the test topic. In the likely event of ties, given that there are potentially hundreds of candidate sentences and only 3 possible scores in the highest case, a Brill

tagger[4] is applied to the text of the facets, and tied sentences are re-sorted by the number of proper nouns contained in the sentence. The motivation of this process is that proper nouns would be included in sentences that are more likely to be relevant, rather than just segue sentences in a news article.

6. Since only novel sentences are scored as vital, it becomes necessary to remove system responses which would map to the same nugget. All sentences with more than 50% non-stopword stems in common with a higher ranking sentence are removed.

7. The top n ranked sentences for each topic are output by the system. $n = 30$ seems to be a standard number of nuggets to return so as not to be too verbose but to allow for enough sentences to be returned that most of the vital nuggets would be accounted for in the sentences returned by the system.

Using the Nugeteer automated evaluation system (described in the next chapter) on the output from the Giraffe system on the 2006 ciQA topics, an F-score of 0.3388 was obtained. Our baseline system got a score of 0.3356, which is close enough to the 2006 system that we can assume they perform as well as each other.

3.2 Wikipedia Link Structure

As we have noted in the previous chapter, Wikipedia is gaining wide acceptance as a source for lexical and semantic information in IR and NLP. Wikipedia is a free encyclopedia which any user can edit and is formatted as hypertext such that the articles contain links to other articles in the encyclopedia.

Each article in Wikipedia effectively describes a concept for the encyclopedia. In Figure 3.4 we can see an example article for the baseball player "Hank Aaron". Such a concept would not be described in other databases, such as WordNet, since it is a multi-word unit, and describes a proper noun. Thus, the coverage in Wikipedia is greater, and is constantly being updated by the thousands of contributors to Wikipedia.

In order to allow users to navigate around Wikipedia, the encyclopedia is written in a hypertext markup so users can browse to articles that represent concepts referenced in other Wikipedia articles.

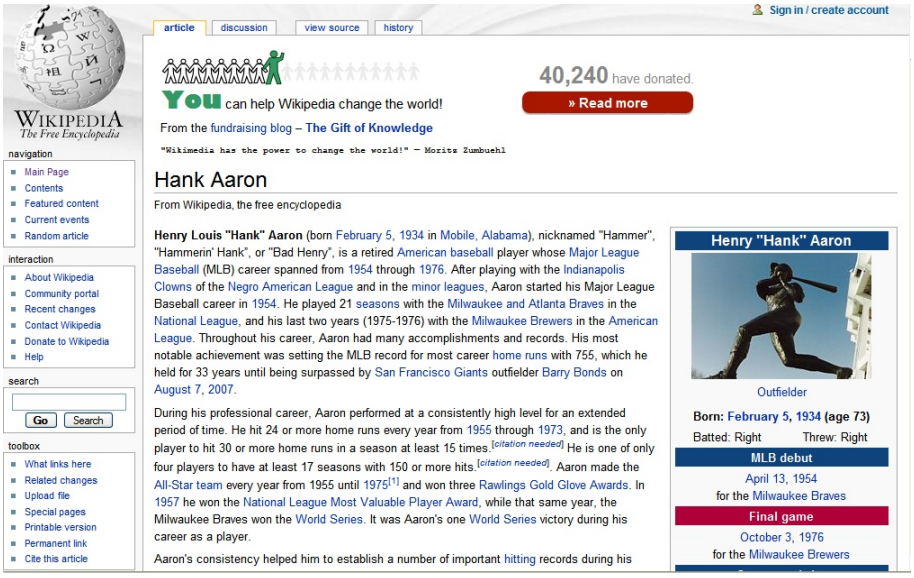

Figure 3.4: The Wikipedia Article on Baseball Player Hank Aaron

As noted in our previous section, Craswell et al used the "anchor description assumption" to improve site-finding[9]. This assumption states that the anchor text of a link describes its target. For example, the text "Internet Movie Database" would accurately describe the document at location `http://www.imdb.com` if it were the anchor text on such a link. Large commercial search engines, such as Google[5] use anchor texts in the manner to improve search results.

By applying this notion to the Wikipedia link structure, we can see that the links to an article can provide additional synonyms for the article if the anchor text on that link is not exactly the same as the article title. This is possible in Wikipedia since there is no requirement that the anchor text match the title of the article it links to.

From the Wikipedia guidelines for what the anchor text should be for a link, we can assume that, provided editors are following the rules, the anchor text of the link will be of high quality. As we can see from this excerpt from the Wikipedia manual of style[2]:

> "It is possible to link words that are not exactly the same as the linked article title, for example, [[English language—English]]. However, make sure that it is still clear what the link refers to without having to follow the link." -Wikipedia Manual of Style

[2] `http://en.wikipedia.org/wiki/Wikipedia:Manual_of_Style_%28links%29`

Looking at Figure 3.5, we can see a diagram of several articles linking to the "United States" article with different articles linking to it with different anchor texts.

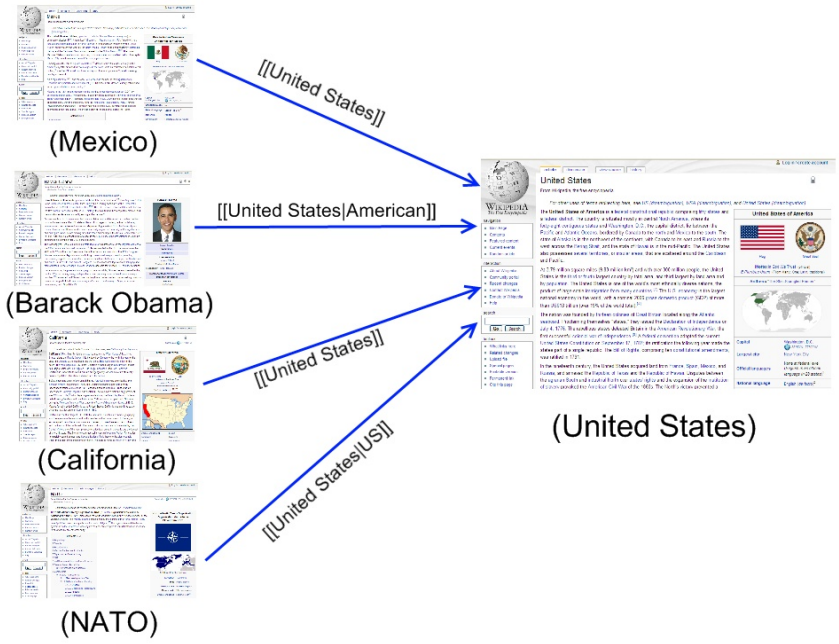

Figure 3.5: Link Structure and Anchor Text of a Wikipedia Article

Looking at the second part of a link in its Wikipedia markup, we can see what the alternate anchor text for a link would be, and thus a synonym for the article title according to the anchor description assumption.

Looking at the complete set of links which point to a single article, we notice that the frequencies of the distinct anchor texts approach a Zipfian distribution. As we can see in Figure 3.6, the United States article has a small number of anchor texts labeling the bulk of the incoming links.

In the next chapter, we will use this diversity in anchor texts to develop a method of synonym detection for the facets of a ciQA topic. Using that method, it will become possible to map facets to Wikipedia articles in order to gain these synonyms from the different anchor texts which point to the article representing the facet.

However, as we will see in the next section, we first need to determine what the best articles would be for the different ciQA topics, given that there often will not be a perfect title match, or a single article for a title match.

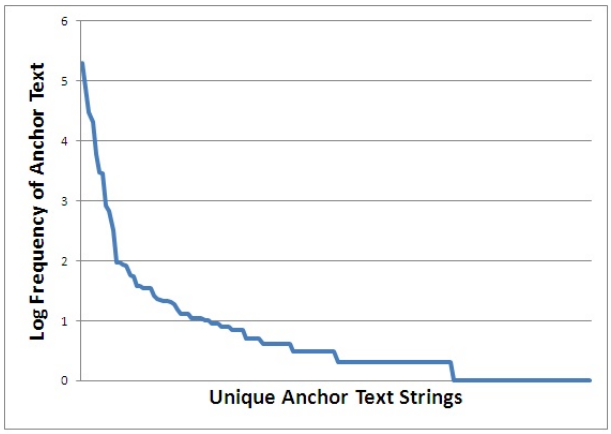

Figure 3.6: Log Frequencies of Anchor Text on Hyperlinks to "United States" Article

3.3 Human Assessors' Selection of Wikipedia Articles

Before we propose an algorithm to find relevant Wikipedia articles for ciQA facets, we should first determine a gold standard of articles we can compare automated results with. It also becomes necessary to find out what level of agreement exists between users in terms of what articles they select for each facet. The more agreement exists, the more confidence we have that an algorithm could perform the task as well as a human assessor.

Starting with a pool of 24 assessors, all of whom are graduate students, we randomly assigned each assessor to one of two groups. The first group performed assessments for the ciQA 2006 topics and the second group performed assessments for the ciQA 2007 topics. Our intent is to show the level of agreement of the assessors picking relevant articles for the facets in the respective topic sets and to create a list of consensus articles which will become our gold standard.

In order for an assessor to determine the relevant articles for a facet, there must be a "short list" for them to pick from in order to make the job of the assessor easier to perform and not requiring an open ended search of the entire Wikipedia collection for relevant documents. In order to facilitate quicker assessment, a short list of up to 10 candidate articles was obtained by taking articles from a Yahoo! search restricted of the `en.wikipedia.org` domain for the facet, title lookups of all combinations of terms in the facets, and expanding disambiguation pages.

The assessors are given the following instructions:

> "For the given facets in the ciQA topics, please check off the MINIMAL set of Wikipedia articles (for which the gloss is provided), which best describes the facet. As a general rule of thumb, tick off the Wikipedia article that would most likely need to be looked up by someone to best understand the concept. If no article best suits the concept, please do not select any articles. For example, last year's **goods in the food-for-oil program** would only require **'Oil-for-Food Programme'** to be selected, not **'Good (economics and accounting)'**. The only time multiple articles should be selected is when there are clear, independent entities, for example, last year's **the Moral Majority or the Christian Coalition** could have both **'Moral Majority'** and **'Christian Coalition of America'** selected. More abstract concepts like **the use of illegal, performance-enhancing substances** should not have any articles selected."

This prompt was given to each assessor at the top of every page of the interaction. Examples of the webpage can be seen if Figures 3.7 and 3.8.

For example, the facet **Illegal Immigrants** would yield the following short list of articles from Wikipedia:

1. Movement Against Illegal Immigration
2. 2006 U.S. immigration reform protests
3. Illegal drug trade
4. Illegalism
5. Illegal immigration
6. Immigration to the United States 7. Immigration
8. History of US immigration
9. Illegal immigration to the United States

When attempting to find agreement between assessors for such tasks, Cohen's Kappa method has been used when there were only two assessors[29]. However, in our case we have multiple assessors amoung which we need to measure agreement, so we choose to use Fleiss' Kappa:

$$\kappa = \frac{\bar{P} - \bar{P}_e}{1 - \bar{P}_e}, \tag{3.4}$$

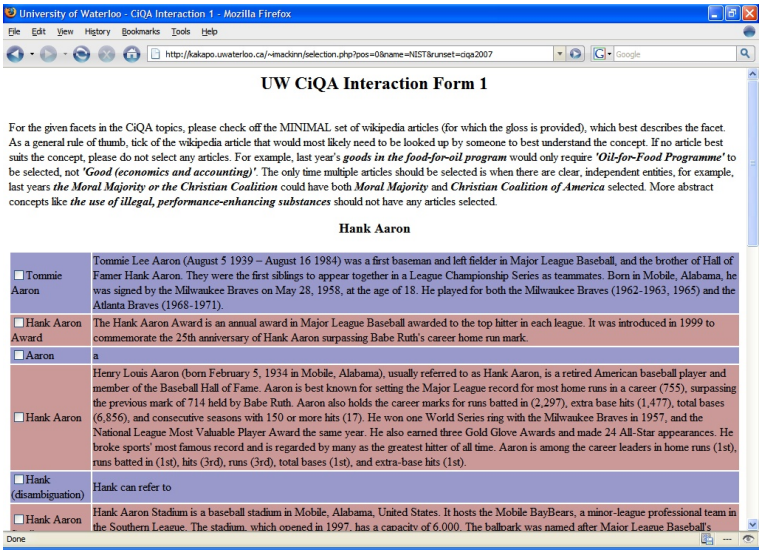

Figure 3.7: Manual Wikipedia Article Selection Screenshot 1

where

$$\bar{P} = \frac{1}{Nn(n-1)}(\sum_{j=1}^{k} N_{ij}^2 - n) \tag{3.5}$$

and

$$\bar{P}_e = \sum_{j=1}^{k}(\frac{1}{Nn}\sum_{i=1}^{N} n_{ij})^2 \tag{3.6}$$

N is the number of trials, in our case $N = 72$ for ciQA2006 $N = 73$ for ciQA2007, both having 30 topics. Some topics contain 3 facets while others contain 2, depending on the query template. In both cases $n = 12$, which is the number of assessors. k depends on how many topics were taken for the short list and is the possible number of permutations of articles selected from that short list. As an upper bound, k can be at most $2^{10} = 1024$. Finally, n_{ij} represents the number of raters who assigned the ith subject to the jth category.

When looking at the agreement scores, we see that the 12 assessors for the ciQA 2006 topics had a Kappa coefficient of 0.4938, while the 12 assessors for the ciQA 2007 topics had a Kappa coefficient of 0.5552. While the generally accepted standards[23] are that anything higher than 0.8 is considered near perfect agreement, we do see a moderate level of agreement from our assessors in both sets, especially considering the large group performing the assessment. Interestingly, the ciQA 2007 topics have a higher level of agreement than ciQA 2006 topics. A possible explanation for this is that the ciQA 2007 topics are based on the AQUAINT-2

Figure 3.8: Manual Wikipedia Article Selection Screenshot 2

corpus, which has news articles from 2004 to 2006, while the ciQA 2006 topics are based on the AQUAINT-1 corpus, which has news articles from 1998-2000. Wikipedia did not exist before 2001, and was considerably more popular during the time spanning AQUAINT-2. It is possible the coverage of topics is more complete for ciQA 2007 than for ciQA 2006, thus it is more likely the assessors could agree on the most appropriate article.

For both ciQA 2006 and ciQA 2007 a set of consensus articles is created by taking the most frequent article selection for each facet. The possibility of no articles being selected for a facet was also an option in addition to any other combination of articles from the short list.

In the next Chapter we will use the consensus articles of the human assessors as a gold standard to create a method to automatically resolve Wikipedia articles for a facet, and integrate that ability into our baseline ciQA system.

Chapter 4

Implementation

4.1 Automatic Selection of Wikipedia Articles

Before we can start to look at the anchor texts that point to articles representing ciQA facets, we must first present our method for automatically selecting articles. As noted in the previous chapter, there can exist a single perfect article for a facet, multiple suitable articles, or none at all.

We have devised a method of using the anchor text within Wikipedia links in order to resolve a small set of concepts which are represented in a candidate sentence. However, in this case we use the anchor text to help find an article rather than find synonyms for it.

The anchor texts which point to the article will contain other phrases which are synonymous for the concept represented in the article. These additional anchor texts are necessary to get a better understanding of what concepts are represented in the text. As we can see in Table 4.1, there are several different anchor texts pointing to the "radio waves" article of varying frequency. We can use these anchor text strings as a dictionary to help us determine which phrase should be mapped to which article in Wikipedia.

However, we can also see the converse is true in that a specific string may exist as anchor text to multiple different articles, as we can see in Table 4.2.

Using the anchor text and their relative frequencies, we define the algorithm to turn a facet into a list of concepts represented by Wikipedia articles as follows:

Anchor Text	Linking Frequency
radio waves	72
radio	4
radio wavelengths	2
airwaves	1
electromagnetic vibration	1
radio signals	1

Table 4.1: Frequency of Anchor Text for "Radio Waves" Article

Article Name	Anchor Text Frequency
radio waves	72
radio frequency	10
Electromagnetic radiation	3
radio	2
Radio Waves (album)	1

Table 4.2: Frequency of Links to Articles that have "Radio Waves" as Anchor Text

1. Set the window length to n.

2. For each possible position of the window, check all anchor text in Wikipedia to see if the phrase or term is recognized. If it is, record the matching string and drop the words covered in the window from future consideration. See Figure 4.1.

3. Decrease the length of the window by one ($n = n - 1$). If the window length is 1, do not look up stopwords in term dictionary, simply ignore. Go to step 2 if window length is greater than 0.

4. For terms extracted from the query, look at the frequency of that term when linking to different articles. If an article has a majority of the links with that term as anchor text pointing to it, resolve that article to be the most relevant article for that multi-word unit. If no article has more than half the links with that anchor text pointing to it, drop the multi-word unit from consideration, as the term is ambiguous. However, if the frequency of anchor text linking to that article is less than 2, it is ignored and dropped from consideration.

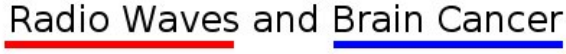

Figure 4.1: Window Recognizes a Multi-word Unit from the Facet.

5. If there are multiple articles resolved for the query, select whichever article has the highest number of incoming links from all other Wikipedia articles to be the most relevant Wikipedia article for the given facet. See Figure 4.2.

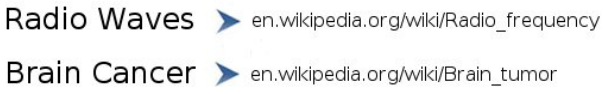

Figure 4.2: Multi-Word Units Resolved to most Frequent Article for Anchor Text

In our experiments, we initially set $n = 5$. Few of the facets are longer than 5 words and those that are tend to represent more abstract ideas which cannot be specified in a single article.

By running this algorithm on the ciQA 2006 and ciQA 2007 test topics, we get sets of articles for every facet in each topic. To compare these automatically retrieved articles with the consensus articles of the human assessors found in the previous chapter we again use Fleiss' Kappa. Looking at this agreement, we find there to be a 0.6206 agreement between the human consensus articles and the automatically retrieved articles for the ciQA 2006 topics, and an agreement of 0.6764 for the ciQA 2007 topics. Both of these coefficients would be considered "substantial agreement" using the informal interpretation given by Landis and Koch[23].

Again we see a greater degree of agreement among the ciQA 2007 data possibly on account of the more time-relevant data in the AQUAINT-2 corpus for Wikipedia.

4.2 Integration of Semantic Enhancements to Base System

In order to test the ability of anchor text to improve nugget retrieval, we must first introduce a method of using the articles we have selected for each facet to be integrated into the base ciQA system described in the previous chapter.

If, for a given facet $\Gamma = \{\gamma_1\gamma_2...\gamma_n\}$ where each γ_i is an individual term in the facet, we have corresponding Wikipedia articles which have anchor text linking to them, the set of anchor text phrases for that facet will be $A = \{\alpha_1\alpha_2...\alpha_n\}$, where each α_i is an anchor text on a link to one of the Wikipedia articles resolved for the facet with a frequency across the Wikipedia corpus greater than 1. Ensuring that

at least 2 articles link to the facet-corresponding one with the same anchor text will prevent potentially vandalized articles from introducing noise into the set of synonyms for the facet, A.

In the ideal situation, only one Wikipedia article is resolved for a facet, with no terms leftover from the facet. In this case, each α_i represents a high-quality phrase which multiple editors on Wikipedia have agreed is a reasonable referent for the concept being described in the linked article. Thus, we can use is it as a substitute for the facet being sought. However, we find that only 45 of the 72 facets, or 62.5%, of the ciQA 2006 facets fit this optimal case.

We modify the baseline system described earlier to incorporate the information from a facet's set of anchor text, A, in addition to the set of terms in the facet, Γ. A candidate sentence, S, would be formally scored as

$$score(S, \gamma_i) = \begin{cases} 1.2 & \text{if at least one of } \alpha_i \in A \text{ exist in S} \\ 1 & \text{if at least one of } \gamma_i \in \Gamma \text{ exist in S, and no } \alpha_i \in A \text{ exist in S} \\ 0 & \text{otherwise} \end{cases} \tag{4.1}$$

The score of 1.2 is rather arbitrary. It just needed to be higher than 1, but low enough such that 2 matches from A would not be ranked higher than 3 from Γ. Different saturation levels were experimented with but none yielded any significant differences.

The problem is that A can be very large for popular articles. Ideally, we want to up-weight a candidates sentence's score if it contains an anchor text phrase from A in it as opposed to simply a term from the facet, but we can't have the size of A be so large that trivial anchor text matches give undue relevance to a candidate sentence. We will explore 3 specific implementations to find A.

The first method, named **WIKI**, will use the above algorithm to resolve a facet to the single most frequently occurring article for anchor text given in the facet. The top 7 occurring anchor texts pointing to that article make up the list of valid synonyms, A. In later chapters, when referring to this method independent of ciQA, the function to take a string of text, resolve it to Wikipedia articles, and extract the top 7 occurring anchor texts will be called "FacetExpand".

In the second scoring method, **WIKI-LIST**, a facet is not mapped to a single article, but rather the entire list of potential articles for a facet is taken into account.

Using the above algorithm, a list of potential articles for a facet is determined. The intuition is that a relevant phrase can be anchor text on links to several different articles. An illustration of this can be seen in Figure 4.3. The anchor texts are weighted by the sum of the frequencies of an anchor text linking to an article multiplied by the popularity of that article, ie. the total number of incoming incoming links.

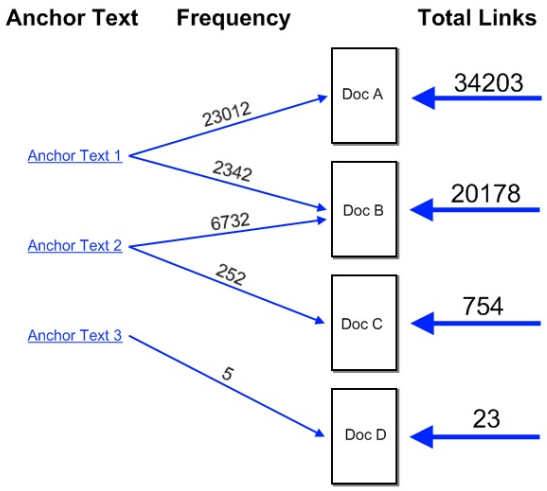

Figure 4.3: Same Anchor Text Pointing to Different Articles

In order to account for this, all anchor texts which point to any of the top articles are considered. Each anchor text, x is assigned a score of:

$$score(x) = \sum_{t \in T} \log(x_t) \cdot \log(\|t\|), \tag{4.2}$$

where T is the set of articles in which the phrase exists as anchor text as a link to the article, x_t is the frequency of x occurring as anchor text on links which point to article t, and $\|t\|$ is the frequency of all incoming links to article t. The set of valid synonyms, A, is determined by the top 7 scoring anchor texts.

In the third method, **WIKI-YAHOO**, the article selection algorithm described above is not used and instead, a top 10 list of potential articles is found using a Wikipedia domain-restricted Yahoo retrieval using the facet as a query. Each anchor text, x is assigned a score of:

$$score(x) = \sum_{t \in T} (10 - rank(t)) \cdot \log(x_t), \tag{4.3}$$

where T is the set of articles in which the phrase exists as anchor text as a link to the article, x_t is the frequency of x occurring as anchor text on links which point to article t, and $rank(t)$ is the ranked position of the document t returned by the Yahoo search engine, and T and x_t are defined above. Again, the set A is determined by the top 7 scoring anchor texts.

Regardless of the specific implementation of sentence score, sentences are then sorted according to score as before. The only difference from the baseline system is the integration of A, the terms from the anchor text. The remaining issue is what method is used to select the articles from Wikipedia for the given facet, for which we described an automatic method earlier.

4.3 Justification for Nuggeteer as an Experimental System

Notwithstanding the changes in the base section using Wikipedia, there are many parameters in the base system from the previous chapter which need to be optimized. Given that the system was created for the first ciQA task, there was not enough data to perform tweaking; however, the ciQA 2006 set now exists as a valid training set.

It is possible to hold all parameters steady and find the best values for a specific parameter by examining the change in ciQA F-scores. However, the only completely accurate way to judge an F-score for a ciQA run is to have human assessors assign system responses to answer key nuggets. This poses a problem for experimentation since the turnaround time for an assessment is long, given the human intervention, and the only official scores are the ones issued by assessors at NIST during TREC. NIST also limits the number of runs a group can submit.

To facilitate the experiments we wish to conduct, an automated way of generating a reliable F-score is necessary. There exist 2 software systems which are designed for this purpose: Pourpre[25] and Nuggeteer[30]. We compare version 0.8 of Nuggeteer and version 1.1 of Pourpre to see which system gives a closer approximation to the actual results for the ciQA 2006 data set given by NIST.

To determine which evaluation system gives the scores closest to that of the human assessors, we calculate the Peason's correlation coefficient of the F-scores as well as the Kendall's Tau score from each of the submitted runs to ciQA 2006

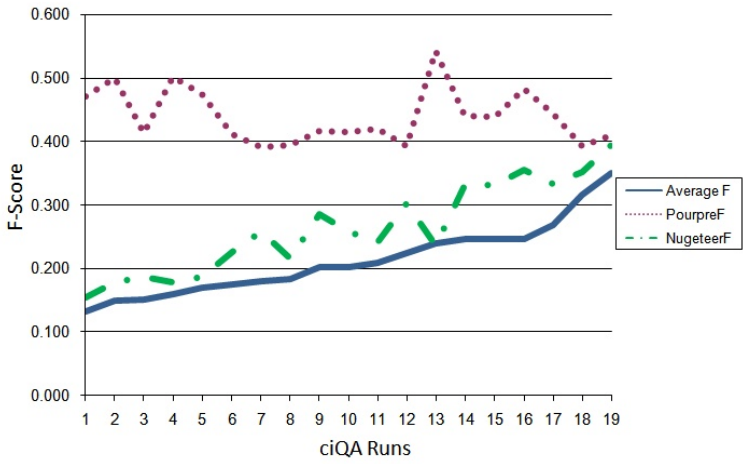

Figure 4.4: Comparison of Human Assessor, Pourpre, and Nuggeteer Score for ciQA 2006

Method	Pourpre (Term Count)	Pourpre (IDF Values)	Nuggeteer
Pearson	0.3267	0.7775	0.8216
Kendall's Tau	0.6101	0.5995	0.7798

Table 4.3: Correlations of Pourpre and Nuggeteer to Official ciQA 2006 Scores

to the score given by their Pourpre and Nuggeteer outputs. In total, 19 runs are compared.

Comparing the official score of each of the submitted runs to NIST with their Pourpre and Nuggeteer scores, we get the results in Table 4.3.

From this we see that Nuggeteer outperforms both methods of Pourpre in both Pearson and Kendall's Tau measures. While Pourpre has been used in previous experiments[26], we clearly see that Nuggeteer offers a higher correlation to official scores; thus, Nuggeteer shall be used in our experiments in this paper whenever an F-score needs to be calculated.

For the purpose of this work, any further F-score will be derived using Nuggeteer unless otherwise stated. The F-scores noted here will also be generated using the binary mechanism described in Chapter 2.

From here, it is now possible to tune the parameters of the base system to ensure that the ciQA system being developed is as high performing as possible.

4.4 Parameter Tuning

4.4.1 Number of Documents

Looking at the methods described in the previous chapter for our base system, the first parameter encountered is the number of documents retrieved using the words in the facets as query terms. Initially, the base system retrieved 200 documents for consideration as sources for information nuggets. However, this number has yet to be justified in any capacity.

In order to find the appropriate number of documents to return from the BM25 retrieval, we look at the results from the ciQA 2006. As part of the evaluation of the 2006 task, NIST released a collection of all the submitted nuggets, the document id they came from, and what their assignment was. Looking at the set of nuggets which have been labeled "vital" and "okay", we can derive a list of documents which we can consider "relevant". For the ciQA 2006 results, a total of 598 relevant documents were found.

By performing a BM25 retrieval for the topics and taking the top n documents, we can see how accurate the retrieval of the 598 documents is in Figure 4.5. Clearly, we see diminishing returns on adding new documents very quickly.

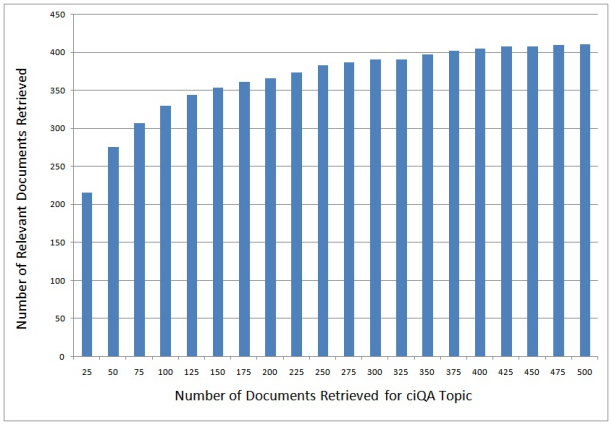

Figure 4.5: Number of Relevant Documents Retrieved for ciQA Topic from 598 Total

Looking at Figure 4.5, we can see how the 200 document retrieval would get a good coverage of the documents from which the vital and okay nuggets were derived, but ultimately, we are interested in improving the F-score of the system. Thus, it become more prudent to tune based on the F-score outcome of the base

system keeping all other parameters of static. We can see the results on testing for different number of documents in the initial retrieval in Table 4.4.

Number of Documents	F-Score
25	0.3194
50	0.3243
75	0.3288
100	0.3242
125	0.3301
150	**0.3354**
175	0.3330
200	0.3259
225	0.3303
250	0.3295
275	0.3293
300	0.3257

Table 4.4: Effect on F-score of Initial Number of Documents Retrieved

It becomes clear from the results in Table 4.4 that fewer retrieved documents would yield slightly better results. However, the differences between the number of documents retrieved seems to make a statistically insignificant change in the F-score, leading us to believe that most 'vital' or 'okay' nuggets are covered in the top documents retrieved rather in the latter documents.

4.4.2 Novelty Threshold

The second parameter we need to investigate is how much overlap between sentences is necessary to regard the two as having the same meaning. According to the ciQA F-scoring method, redundant pieces of information do not contribute to the score and only lengthen the system‘s response, causing a run of diminished value.

As noted by Lin et al[26], this mirrors the TREC novelty tracks in that the problem is to only return novel pieces of information.

The base system uses a simple method of removing any response sentence which contains more than 50% of the same non-stopword stems as any perviously returned sentence. A more accurate measure could be used by using the same language modeling approach as the Nuggeteer system[30]; however, using the method in checking for redundancy and evaluation would lead to a bias in the experiments. Thus, we tune the threshold parameter rather than replace the method altogether.

The parameter being tested in this section is what threshold percentage of non-stopword stems in common should two sentences have before they are considered containing the same information. Were a threshold set too high, all would be declared novel, and thus the same vital nugget could be returned many times. Conversely, if the threshold were set too low, it becomes possible that two sentences containing information from two distinct vital nuggets could be lost due to the system thinking both articulate the same piece of information.

To find the optimal overlap, we use the baseline system from the ciQA 2006 track and retrieve the top 100 sentences returned for each topic for each of the systems. Each of the different sentence similarity methods will be used to take these 100 sentences for each topic, remove redundant sentences, and then return the top 30 sentences to the Nuggeteer system for evaluation.

The following algorithm will be used to perform this operation:

Algorithm 1 NuggetOverlap(K - ordered list of nuggets of length,$n = 100$, τ - similarity threshold, $m = 30$ - number of nuggets to return)

```
i = 0
for i < n do
  j = 0
  overlap = False
  for j < i do
    if sim(K_i, K_J) > τ then
      overlap = True
      j++
    end if
  end for
  if !overlap then
    Append K_i to K'
  end if
  if ||K'|| > m then
    return K'
    i++
  end if
end for
```

This method ensures that all sentence similarity thresholds are on equal footing for their evaluation and thus other factors are not affecting the comparison. As we can see in Table 4.5, we get a slightly better performance when we use a threshold for non-stopword stems of 60% rather than the original 50%.

Percent	F-Score
0.1	0.0690
0.2	0.1345
0.3	0.2623
0.4	0.3161
0.5	0.3327
0.6	**0.3425**
0.7	0.3313
0.8	0.3324
0.9	0.3239

Table 4.5: Setting Threshold for Novel Sentences

4.4.3 Number of Returned Sentences

The last parameter we will investigate is the number of candidate sentences returned by the system. The original system returned a static 30 sentence responses for every ciQA topic; however, the actual number of sentences retrieved will have a large impact on the F-score of the system. A verbose number of responses are penalized as a surrogate for precision in the F-score calculation and too few responses will lead to not enough vital nuggets being returned to garner a top F-score.

Again, to test for this, we simply hold all other parameters in the baseline ciQA system static while varying the number of sentences given as responses. Conceivably, any sentence that is returned when n sentences are returned will also be contained in the system responses for more than n sentences returned, given that the number of sentences returned does not affect their ordering. Thus, when looking at the number of sentences the system returns, there should be a point where the penalty for verbosity outweighs any chance of finding a deeper buried vital nugget.

As we can see from Table 4.6, a slightly higher 50 sentences can be returned by the system before we start to see diminishing returns.

Having proposed a Wikipedia-based synonym expansion method and integrated it into our baseline ciQA system, we can now use the enhanced system along with the optimal parameters discovered to experiment with the ciQA evaluations from 2006 and 2007. However, in the next chapter we will also explore how the "FacetExpand" method can be used in document retrieval and the synonym detection tasks outlined in Chapter 2.

Number Returned	F-Score
60	0.3304
55	0.3349
50	**0.3442**
45	0.3435
40	0.3418
35	0.3325
30	0.3284
25	0.3201
20	0.2882

Table 4.6: F-scores for Number of Nuggets Returned

Chapter 5

Experiments

5.1 Document Retrieval

The first evaluation we will perform is comparing a baseline BM25 retrieval of the 2005 TREC HARD Track, a Wikipedia-based query expansion method, and other more common query expansion methods for the first 25 queries from the TREC 2005 Hard Track.

In order to gauge the performance of terms extracted from Wikipedia anchor texts, we first need to establish a baseline. The first 25 queries from the TREC 2005 HARD Track will be used as a test of this method. Using BM25 document retrieval, the title terms are used in the search to rank the top 100 documents.

Using the article resolution algorithm described in Chapter 3, we take the first 25 HARD 2005 queries and apply our "FacetExpand" method to the query text. In the run we designate "HARD-WIKI", the original query terms are up-weighted by 3.5, and the new expansion terms are included in the BM25 retrieval of 100 documents. Using **trec_eval**, we are able to see the results of the baseline runs the expansion in Table 5.1.

Expansion Type	Map	bpref	P@10
Baseline	0.1955	0.2303	0.4280
HARD-WIKI	0.1788	0.2250	0.400

Table 5.1: Effect on Document retrieval of FacetExpand

As we can see, this method actually degrades performance of the retrieval on all the standard measures. However, as Diaz and Metzler pointed out, using external

corpora as a source for terms may be useful[12]. Instead of looking at the anchor text, we now look at the terms which actually exist within a Wikipedia article. To accomplish this, we use the same procedure outlined in Chapter 3 to select an article; however, instead of pulling synonyms from anchor text, we take the top 10 terms from the article according to their TFIDF value. We call the BM25 retrieval with expansion terms derived from the terms in the automatically selected Wikipedia articles HARD-WIKI-A

We now use several other query expansion methods to test the performance of Wikipedia articles as a source. We are more interested in contrasting the quality of the terms selected rather than the expansion procedure itself, so for our experiments all original query terms will have their weights multiplied by 3.5 in the BM25 search, similar to other comparisons[21].

We do several runs in addition to the baseline to test the terms selected from different query expansion methods. For each method we extract ten terms for the query expansion. These query expansion methods are: assuming the top 10 documents are relevant and taking terms from those documents (BM25), the most relevant Wikipedia page (HARD-WIKI-A), the top ranked Wikipedia article chosen by performing a Yahoo! search of the original query terms restricted to the `http://en.wikipedia.org` domain (HARD-WIKI-YAHOO), and the Wikipedia article which is selected by human judges (HARD-WIKI-HUMAN).

The HUMAN-WIKI run was done by giving 5 human operators the TREC queries with instructions to find the most relevant Wikipedia article which described as much of the query as possible. The article with the largest number of judges declaring it relevant was used for expansion.

Expansion Type	Map	bpref	gmap
Baseline	0.1955	0.2303	0.1013
BM25	0.2209	**0.2814**	0.0752
HARD-WIKI-A	0.2093	**0.2689**	0.0635
HARD-WIKI-YAHOO	0.1809	0.2500	0.0535
HARD-WIKI-HUMAN	0.2309	**0.2933**	0.1209

Table 5.2: Effect on Document Retrieval of Selecting Terms from Wikipedia Articles

As we can see from the experiment results in Table 5.2, the performance of query expansion with terms from an automatically selected Wikipedia article by our algorithm performed better than if the articles were selected by the Yahoo

search engine, but not as well as traditional blind relevance feedback which draws expansion terms from the top n documents, which are assumed to be relevant.

In the case where the articles are selected by human operators, we see great improvement in the performance, with even greater stability. This method is more akin to standard relevance feedback where a human operator selects the terms for feedback to the system. However, it does show the potential for improvement by using Wikipedia as a source for query expansion terms.

5.2 ciQA 2006 and ciQA 2007

The major objective of our synonym finding was to improve ciQA nugget retrieval. Using the Wikipedia-enhanced methods described in the previous chapter, we can compare the various weighting schemes against the baseline from the previous year.

The ciQA 2006 topic set draws their answers from the AQUAINT-1 corpus while the ciQA 2007 topics come from the AQUAINT-2 corpus. Aside from that change in corpus, the same methods are used on both the 2006 and 2007 set. Given that most development was tested with the 2006 set, the 2007 set becomes a better evaluation set.

Evaluating the returned sets using the Nuggeteer binary evaluation described earlier, we have the results for the 2006 and 2007 ciQA F-scores for the baseline, WIKI, WIKI-LIST, and WIKI-YAHOO adaptations in Table 5.3.

Method	2006 F-score	Improvement	2007 F-score	Improvement
Baseline	0.3356	n/a	0.3388	n/a
WIKI	0.3718	10.8%	**0.3663**	8.1%
WIKI-LIST	0.3625	8.0%	0.3631	7.2%
WIKI-YAHOO	**0.3759**	12.0%	0.3556	5.0%

Table 5.3: F-Scores for 2006 and 2007 ciQA Runs

Looking at the individual results of the 30 topics for ciQA 2007, we find that the automatic article selection improves F-scores in 8 of the topics, leaves 20 static (less than 2% change), and decreases 2. While the 2006 improvements were slightly better than the 2007 improvements, we do see that the Wikipedia-enhanced nugget retrievals give a modest improvement in the F-scores of the task. All of the Wikipedia-enhanced methods give some level of improvement; however, the different topic sets

yield different best-performers. The WIKI-YAHOO method fared the best in 2006 while the standard WIKI gave the best performance for the 2007 set.

Looking at the WIKI method, we are also interested in seeing what effect the articles selected for each facet has on the outcome of the nugget retrieval. In order to investigate this, we can compare the WIKI method when it uses the automatically selected articles versus WIKI when it uses the consensus articles from the human assessors. Running the 2006 and 2007 topics on the 2 WIKI adaptations, one for automatic article selection, and one for human article selection, against the baseline, we can see what impact the article has on the retrieval.

Run	2006 F-score	Improvement	2007 F-score	Improvement
Baseline	0.3356	n/a	0.3388	n/a
WIKI (automatic)	0.3718	10.7%	0.3663	8.1%
WIKI (consensus)	0.3722	10.9%	0.3676	8.2%

Table 5.4: Comparison of ciQA F-Scores for Human and Automatically Selected Articles

In Table 5.4 we can see a modest improvement in F-scores using the two Wikipedia-expansion methods. While the agreement between the consensus articles and the automatically selected articles for the 2006 and 2007 ciQA topics were only 0.6202 and 0.6764, respectively, the resulting F-scores of the methods are virtually the same. This means that the performance gains of this method are not dependent on a large user group assessing relevance of Wikipedia articles and can be obtained using the automatic method we have described.

As we saw by looking at the individual WIKI results, the gains appear to come from a few, well-selected articles for facets which expand the high-quality synonyms. There are few that perform worse, but it could also be that the pertinent articles for the best-performing topics were contained in both the consensus set and the automatically-selected set.

Looking more closely at the 2007 topics, the topics which WIKI gained the best improvement were:

> What evidence is there for transport of [automobiles] from [China] to [Russia]?

> What effect does [glucosamine] have on [arthritis]?

In the first example, it becomes clear that the "automobiles" article would have the synonyms "car" and "cars" as anchor text on many links pointing to it. The inclusion of "cars" as a meaning for "automobiles" gives it the more common term which would more likely be used in a news article.

In the second case, "glucosamine" was expanded to include the other chemical names for the drug, as well as the names the drug would be marketed under. Again, the inclusion of such high-quality synonyms which would be more likely to appear in a news article make the method very helpful.

Looking at the 2007 topics for which WIKI lowered the resulting F-score, we see:

> What evidence is there for transport of [illegal immigrants] from [Croatia] to [the European Union]?
>
> What effect does [the Red Tide] have on [sea creatures]?

In the first case, the "the European Union" article had little in terms of diversity of link anchor text. The majority of relevant nuggets would make reference to member nations within the European Union political entity, but not the body itself. To make better use of this, the category page for the European union could potentially be used to see which member nations could be considered a stand-in for the "European Union" facet.

In the second case, "sea creatures" resolved to the "Marine Biology" article due to a small number of links with "sea creatures" as anchor text to that article. There was just enough links to "Marine Biology" with the "sea cratures" anchor text to be considered sufficient for a synonym. The rest of the synonyms for "Marine Biology" caused query drift away from the actual animals living in the sea, to the broader topic of marine biology.

5.3 TOEFL Test

While the Wikipedia synonym expansion makes modest improvements to ciQA nugget-retrieval tasks, we also wish to see how well the method works when attempting to find synonyms in the general case. As we saw in Chapter 2, the most common test for synonym finding is the TOEFL test introduced by Dumais and

Landauer[22]. Another approach introduced by Bhat et al. was to create a set of chemical pairs with two meanings for the same compounds[3]. In this section, we will evaluation our synonym expansion method using the TOEFL test, and in the next section the Chemical nomenclature test.

Given our article selection and term expansion algorithm, we must make small adjustments to the process in order to allow the Wikipedia-synonym finder to solve this problem:

Algorithm 2 TOEFLSelect(Problem Word P, Potential synonyms $S_{1...4}$, Corrected Sense S_i)

```
wikiset = FacetExpand(P)
for all potential w in wikiset do
  if w in  then
    S_j = w
    break
  end if
end for
if wikiset =NULL then
  "NONE RESOLVED"
else if S_j = NULL then
  output "NONE FOUND"
else if S_j = S_i then
  output "CORRECT"
else
  output "INCORRECT"
end if
```

FacetExpand is a function that maps a term to a Wikipedia article and retrieves an ordered list of anchor texts according to the frequency of the anchor text labeling a link to that article, as outlined in chapter 4 for the WIKI ciQA run. This procedure is repeated for each of the 80 questions in the TOEFL problem set. An output of "CORRECT" indicates a positive detection for the WIKI method, while "INCORRECT" indicates that the WIKI method failed to select the correct sense for the problem word. An output of "NONE RESOLVED" indicates that no article could be resolved from the problem word, whereas "NONE FOUND" means none of the potential synonyms were found in the set of anchor text returned by FacetExpand. Results are tallied to give a percent-accuracy of the Wikipedia-expansion method.

This method is meant to enable some form of comparison between existing

synonym detection systems using TOEFL and the Wikipedia-based method we have proposed. The initial scores given by the work by Dumais and Landauer were 64%[22], while the most recent advances by Turney got a score of 97.5%[39].

Applying this WIKI implementation of a general synonym finder to the TOEFL dataset gives rather dismal results though. Only in 4 cases was the correct synonym detected, giving it a score of only 5%. 21 terms, or 26.25%, didn't even resolve to a Wikipedia article.

The 4 terms that performed correctly were: "infinite", "verbally", "physician", and "construction". However, there were no cases in which "INCORRECT" was returned. In the cast majority of cases, an article was resolved, but all of the anchor text which pointed to it were other lexicalizations rather than synonyms.

This test identifies one of the major shortcomings of using Wikipedia hyperlinks for synonym data, in that dictionary terms are not linked as frequently as other, more significant terms which have developed articles. The majority of the problem words in the TOEFL set are simple nouns and verbs which are common, and would likely not have many internal Wikipedia links pointing to them to assist users who may not understand their meaning. Thus, there is minimal breadth in the anchor text which are used on the links. In some cases, the term isn't even significant to have its own article within Wikipedia.

5.4 Chemical Nomenclature Test

As we saw in Chapter 2, Bhat et al tried a 2-stage LSA approach to the problem[3]. The first test used by Bhat et al. was to get a list of synonym pairings of chemical compounds.

For example, the staring term "Methanal" would have a target synonym "Formaldehyde". Similarly, "Methylpropene" would have the target synonym "Isobutene".

In the method proposed by Bhat et al, a list of similar words is returned by their 2-stage LSA approach and the ranking of the target synonym was found in the listing of similar words and given a rank equivalent to the position of the target word in the returned list. Using the 2-stage LSA approach, an average rank of 28 was found. This average rank did not take into account lists for which the target synonym was not found.

Given that one of the best performing ciQA topics from 2007 was the topic which contained the "glucosamine" facet, we expect that the WIKI approach to synonym detection should perform very well. Again, using the "FacetExpand" function described in the previous section, we can use the WIKI synonym detection method for this test as follows:

Algorithm 3 calculateRank(Start Word S, Target Synonym T)

```
wikiset = FacetExpand(P)
wikiset = wikiset / S
for w_i in wikiset do
  if w_i = T then
    return "Term Found at position " + i
  end if
end for
return "NONE FOUND"
```

Repeating this algorithm for each of the word pairs in the nomenclature we can determine an average rank which can be compared to the method proposed by Bhat et. al, since FacetExpand would similarly return a list of potential synonyms in order.

We can see the results in Table 5.5, comparing the anchor text rank returned by our FacetExpand method with that of the Stage 2 LSA method.

It becomes evident that for cases where the synonym does exist as anchor text pointing to the target article for the chemical compound, the synonym is ranked very high. However, in many cases the article describing the chemical compound has a small number of incoming links and did not have the diversity necessary to have the synonyms.

However, in many cases the problem word was resolved to an article which contained the target synonym in the gloss of the Wikipedia article, rather than as separate anchor text.

Similar to the TOEFL test, when an article is resolved for the problem term, the output is most often positive. However, an article is not always resolved.

Name	Target Synonym	Anchor Text Rank	Stage 2 Rank
Methanal	Formaldehyde	1	2
Ethanal	Acetaldehyde	1	13
Propanal	Propionaldehyde	1	7
Butanal	Butyraldehyde	1	10
Propanone	Acetone	1	15
Ethene	Ethylene	1	11
Propene	Propylene	1	14
Ethenyl	Vinyl	1	76
Propyne	Acetylene		47
Methanol	Methyl alcohol	2	89.5
Ethanol	Ethyl alcohol		58
2-Butanone	Ethyl methyl ketone	2	12.3
2-Propenyl	Allyl		45
Aminobenzene	Aniline		4
Hydroxybenzene	Phenol		1
Phenylmethanal	Benzaldehyde		3
Pentanal	Veraldehyde		11
Dichloromethane	Methylene chloride	1	16
Nonanal	Nonylaldehyde		14
Pentanedial	Glutaraldehyde		15
Cyclohexene	Tetrahydrobenzene		
Methylpropene	Isobutene	1	1
Bromocyclohexane	Cyclohexyl bromide		
Nitromethane	Nitrocarbol		21

Table 5.5: Rank Results for Chemical Compounds test

Chapter 6

Conclusions

6.1 Conclusions

In this work we showed the results of a user study to find the most relevant Wikipedia articles for the components of a ciQA query. Using this data, we found that the article selection by human assessors for the ciQA 2007 topics had a higher level of agreement than the ciQA 2006 topics. We suspect this is largely because the topics from the ciQA 2007 task are derived from the AQUAINT-2 corpus, which contains articles from 2004-2006. Wikipedia was active in this time, whereas the topics for the ciQA 2006 task were derived from AQUAINT-1, which contains articles that predate Wikipedia.

We also proposed an algorithm to automatically select a small set of relevant Wikipedia articles for each facet of a ciQA query. This method was found to have a substantial amount of agreement with the consensus of the human assessors. However, the level of agreement was not high enough to suggest that resolving Wikipedia articles is a trivial task.

This method was integrated with an already high-performing baseline system in order to provide a list of high-quality synonyms to assist with nugget retrieval. Several methods of integration were explored.

We have also shown that using correlations to the official ciQA assessments for 2006, Nuggeteer is the most accurate automatic method of assessing nuggets for the Complex Interactive Question Answering track. Prior to this, no verification of the Nuggeteer and Pourpre systems had been performed on the ciQA task. Using

this software, we were able to show a modest improvement in F-scores for ciQA topics which used the Wikipedia anchor text method of query expansion.

The Wikipedia-enhanced version of the ciQA system gave a modest improvement over the baseline system. Other retrieval evaluations such as TOEFL and chemical nomenclature test were performed with mixed results. The TOEFL tests did not perform well given that the TOEFL test set concentrated on more "dictionary" terms than encyclopedic ones. The coverage of simple verbs is sparse in Wikipedia and did not lend itself well to that evaluation. However, for determining synonyms for chemical compounds, the Wikipedia anchor text methods performed very well. The coverage of science articles in Wikipedia is very broad and allowed for high performance.

6.2 Future Work

6.2.1 Resolution Algorithms

This line of research introduces several new directions involving Wikipedia, which has shown itself to be an up and coming source for lexical information. The first is the resolution of articles from a query. We showed that many previous approaches looked at the selection of a large array of articles for traditional latent semantic analysis. However, our approach is close to ones involving WordNet, in that a small set of lexical data is sought. When trying to resolve an article for a given phrase, there are many interesting questions, such as disambiguation of multiple articles with similar titles and whether a term is significant enough to warrant resolving to an article. We hope to improve our article resolution algorithm by incorporating a part-of-speech tagger and word sense disambiguation tools to more accurately select articles.

Further work could also be done to fine-tune the procedure for extracting synonyms for articles by looking at anchor text. The current method of only taking anchor text which labels a link to an article with a frequency higher than 1 was mostly done because a lack of ciQA datasets meant that there could be no effective training set. Once more sets become available, statistical models could be found to give the most appropriate synonyms based on the distribution of the anchor text.

Expanding the link structure to include the newer Wikipedia property Wik-

tionary, a "wiki-based Open Content dictionary"[1], may also allow for better coverage of simple verbs and nouns which are not well represented in Wikipedia proper.

6.2.2 Connectionist Model of Wikipedia

Connectionist models of cognitive science stipulate that human cognition is ultimately performed by an "interconnecting network of simple units". This model is meant to be analogous to the neurons which exist in the human brain. A spreading activation implies that a unit which has been activated will generate action in all the units to which it is connected. The most classic example of this in computing is the neural network machine learning method.

In the future we hope to begin looking at a connectionist model of Wikipedia articles, treating every link in the corpus as a semantic link between two concepts. Previous work has been performed using a spreading activation model to assist with information retrieval[10], but none have yet used the Wikipedia link structure as a basis for the semantic network on which spreading activation can occur.

Clearly, weights on the links would depend on the strength of the semantic bond between two concepts. Using this method it may be possible to retrieve a list of high-quality related terms which could also be used to aid in nugget retrieval. More importantly, it could be used to find intersections of related terms between two facets.

[1] `http://www.wiktionary.org/`

Glossary

Anchor Text - The string of clickable text associated with a hyperlink. Should give contextual information about the document it points to.

AQUAINT-1 - Collection of news articles from the Associated Press, New York Times, and Xinhua News Agency (English version) from 1996 to 2000.

AQUAINT-2 - Collection of news articles from the Associated Press, New York Times, Xinhua News Agency (English version), Agence France-Presse (English version), China News Agency (English version), and the Los Angeles Times from 2004 to 2006.

BM25 - Ranking function used to give a document a score based on the probability of relevance to a given query.

ciQA - *complex interactive Question Answering*, sub-task of the TREC QA track which has systems return nuggets of information relevant to the relationship between entities.

F-score - Metric for evaluating the output of a ciQA system by taking into account the relevance of the returned nuggets and the verbosity of the output.

FacetExpand - Function described in this work to take a string of text, resolve it to Wikipedia articles, and extract the top 7 occurring anchor texts pointing to those articles.

IR - *Information Retrieval*, the field of searching for information within documents or for documents themselves.

MAP - *Mean Average Precision*, the mean over many retrieval trials's average number of relevant documents returned.

NIST - *National Institute for Standards and Technology*, agency of the United States government with mission to "promote U.S. innovation and industrial competitiveness by advancing measurement science, standards, and technology". Headquartered in Gaithersburg, Maryland

NLP - *Natural Language Processing*, field studying problems of computationally understanding human languages. Sub-field of Artificial Intelligence.

Nugget - Short string of text meant to be given as a response to a query.

Nuggeteer - A "tool for evaluating TREC definition and relationship questions, the AQUAINT opinion questions, and complex interactive question answering (ciQA), all of which can be described as nugget-based tasks."[2]

P@n - The percentage of ordered documents returned in the top n of a retrieval which are considered relevant to the given query.

Pourpre - Scoring script for automatically evaluating answers to complex questions, given an answer key.

Stopword - Function word of insignificant information content. Examples: "of", "and", "the", etc.

TFIDF - *Term Frequency/Inverse Document Frequency*, function to assign numeric weight to a term representing its information content based on the number of times it occurs within a document and the inverse of the number of documents

[2] `http://people.csail.mit.edu/gremio/code/Nuggeteer`

which it occurs in the corpus.

TOEFL - *Test Of English as a Foreign Language*, test used to evaluate an individuals capacity to use American English. Most common use in this work is for its section which quizzes candidates on their ability to find a synonym for a given problem word.

TREC - *Text REtrieval Conference*, hosted by NIST every year, focusing on different IR research areas. Its purpose is to support and encourage research in the IR discipline.

QA - *Question Answering*, branch of IR research dealing with the problem of retrieving answers to questions posed in natural language.

Wikipedia - Online, collaborative, free encyclopedia operated by the Wikimedia Foundation. Launched in 2001 by Jimmy Wales and Larry Sanger, it is the largest encyclopedia on the internet. Can be downloaded in its entirety for individual use.

List of References

[1] Sisay Fissaha Adafre and Maarten de Rijke. Discovering missing links in wikipedia. In *LinkKDD '05: Proceedings of the 3rd international workshop on Link discovery*, pages 90–97, New York, NY, USA, 2005. ACM. 21

[2] D Ahn, V Jijkoun, G Mishne, K Miller, M. de Rijke, and S. Schlobach. Using wikipedia at the trec qa track. In *In: Proceedings TREC 2004*, 2004. 21

[3] James Allan, Courtney Wade, and Alvaro Bolivar. Retrieval and novelty detection at the sentence level. In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 314–321, New York, NY, USA, 2003. ACM Press.

[4] Michel Galley And. Improving word sense disambiguation in lexical chaining, 2003.

[5] Vinay Bhat, Tim Oates, Vishal Shanbhag, and Charles Nicholas. Finding aliases on the web using latent semantic analysis. *Data Knowl. Eng.*, 49(2):129–143, 2004. 10, 56, 57

[6] Bodo Billerbeck and Justin Zobel. Questioning query expansion: an examination of behaviour and parameters. In *ADC '04: Proceedings of the 15th Australasian database conference*, pages 69–76, Darlinghurst, Australia, Australia, 2004. Australian Computer Society, Inc.

[7] Matthew W. Bilotti, Boris Katz, and Jimmy Lin. What works better for question answering: Stemming or morphological query expansion? In *Proceedings of the Information Retrieval for Question Answering (IR4QA) Workshop at SIGIR 2004*, Sheffield, England, 2004.

[8] Eric Brill. Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. *Computational Linguistics*, 21(4):543–565, 1995. 28, 32

[9] Sergey Brin and Lawrence Page. The anatomy of a large-scale hypertextual web search engine. *Comput. Netw. ISDN Syst.*, 30(1-7):107–117, 1998. 33

[10] Chris Buckley, Gerard Salton, James Allan, and Amit Singhal. Automatic query expansion using SMART: TREC 3. In *Text REtrieval Conference*, pages 0–, 1994.

[11] A. Budanitsky. Semantic distance in wordnet: An experimental, application-oriented evaluation of five measures, 2001.

[12] Bullinaria, A. John, Levy, and P. Joseph. Extracting semantic representations from word co-occurrence statistics: A computational study. *Behavior Research Methods*, 39(3):510–526, August 2007. 15

[13] Razvan Bunescu and Marius Pasca. Using encyclopedic knowledge for named entity disambiguation. In *Proceedings of the 11th Conference of the European Chapter of the Association for Computational Linguistics (EACL-06), Trento, Italy*, pages 9–16, April 2006. 21

[14] Zheng Chen, Shengping Liu, Liu Wenyin, Geguang Pu, and Wei-Ying Ma. Building a web thesaurus from web link structure. In *SIGIR '03: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 48–55, New York, NY, USA, 2003. ACM Press.

[15] Kenneth Ward Church and Patrick Hanks. Word association norms, mutual information, and lexicography. In *Proceedings of the 27th annual meeting on Association for Computational Linguistics*, pages 76–83, Morristown, NJ, USA, 1989. Association for Computational Linguistics. 12

[16] C. L. A. Clarke, G. V. Cormack, M. Laszlo, T. R. Lynam, and E. L. Terra. The impact of corpus size on question answering performance. In *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 369–370, New York, NY, USA, 2002. ACM Press.

[17] Nick Craswell, David Hawking, and Stephen Robertson. Effective site finding using link anchor information. In *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 250–257, New York, NY, USA, 2001. ACM Press. 2, 25, 33

[18] F Crestani. Application of spreading activation techniques in information retrieval. *Artificial Intelligence Review*, 11(6):453–482, 1997. 62

[19] James R. Curran and Marc Moens. Improvements in automatic thesaurus extraction. In *Proceedings of the ACL-02 workshop on Unsupervised lexical acquisition*, pages 59–66, Morristown, NJ, USA, 2002. Association for Computational Linguistics.

[20] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41(6):391–407, January 1999. 8, 15

[21] Mona Diab and Philip Resnik. An unsupervised method for word sense tagging using parallel corpora. In *ACL*, pages 255–262, 2002.

[22] Fernando Diaz and Donald Metzler. Improving the estimation of relevance models using large external corpora. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 154–161, New York, NY, USA, 2006. ACM Press. 52

[23] Stephen Dill, Nadav Eiron, David Gibson, Daniel Gruhl, R. Guha, Anant Jhingran, Tapas Kanungo, Sridhar Rajagopalan, Andrew Tomkins, John A. Tomlin, and Jason Y. Zien. Semtag and seeker: bootstrapping the semantic web via automated semantic annotation. In *WWW '03: Proceedings of the 12th international conference on World Wide Web*, pages 178–186, New York, NY, USA, 2003. ACM.

[24] Susan Dumais, Michele Banko, Eric Brill, Jimmy Lin, and Andrew Ng. Web question answering: is more always better? In *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 291–298, New York, NY, USA, 2002. ACM Press.

[25] Hui Fang and ChengXiang Zhai. Semantic term matching in axiomatic approaches to information retrieval. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 115–122, New York, NY, USA, 2006. ACM Press.

[26] Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, and Eytan Ruppin. Placing search in context: the concept revisited. In *WWW '01: Proceedings of the 10th international conference on World Wide Web*, pages 406–414, New York, NY, USA, 2001. ACM. 22, 24

[27] J. R. Firth. A synopsis of linguistic theory 1930-55. 1952-59:1–32, 1957. 12

[28] Evgeniy Gabrilovich and Shaul Markovitch. Overcoming the brittleness bottleneck using wikipedia: Enhancing text categorization with encyclopedic knowledge. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence*, pages 1301–1306, Boston, MA, 2006. 2

[29] Evgeniy Gabrilovich and Shaul Markovitch. Computing semantic relatedness using wikipedia-based explicit semantic analysis. In *Proceedings of The Twentieth International Joint Conference for Artificial Intelligence*, Hyderabad, India, 2007. 2, 23, 24

[30] Jim Giles. Internet encyclopaedias go head to head. *Nature*, 438(7070):900–901, December 2005. 2, 21

[31] Sanda M. Harabagiu. Deriving metonymic coercions from WordNet. In Sanda Harabagiu, editor, *Use of WordNet in Natural Language Processing Systems: Proceedings of the Conference*, pages 142–148. Association for Computational Linguistics, Somerset, New Jersey, 1998.

[32] Daqing He and Yefei Peng. Comparing two blind relevance feedback techniques. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 649–650, New York, NY, USA, 2006. ACM Press.

[33] M. Hearst. Improving full-text precision on short queries using simple constraints, 1996.

[34] Marti Hearst. Multi-paragraph segmentation of expository text. In *32nd. Annual Meeting of the Association for Computational Linguistics*, pages 9–16, New Mexico State University, Las Cruces, New Mexico, 1994.

[35] Richard P. Honeck. Semantic similarity between sentences. *Journal of Psycholinguistic Research*, 2(2):137–115, 1973.

[36] Nancy Ide and Jean Véronis. Introduction to the special issue on word sense disambiguation: the state of the art. *Comput. Linguist.*, 24(1):2–40, 1998.

[37] Mario Jarmasz and Stan Szpakowicz. Roget's thesaurus and semantic similarity. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP-03)*, pages 212–219, Borovets, Bulgaria, 2003. 16, 17, 18, 24

[38] Jing Jiang and ChengXiang Zhai. Exploiting domain structure for named entity recognition. In *Proceedings of the Human Language Technology Conference of the NAACL, Main Conference*, pages 74–81, New York City, USA, June 2006. Association for Computational Linguistics.

[39] Diane Kelly and Jimmy Lin. Overview of the TREC 2006 ciQA task. *SIGIR Forum*, 41(1):107–116, 2007. 1, 5, 29

[40] Adam Kilgarriff. SENSEVAL: An exercise in evaluating word sense disambiguation programs. In *Proceedings of the International Conference on Language Resources and Evaluation (LREC)*, pages 581–588, Granada, Spain, 1998.

[41] Jon M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, 1999. 25

[42] Giridhar Kumaran and James Allan. Umass at trec ciqa. In *Proceedings of the 15th Text REtrieval Conference (TREC 2006)*, 2006.

[43] Adenike M. Lam-Adesina and Gareth J. F. Jones. Applying summarization techniques for term selection in relevance feedback. In *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 1–9, New York, NY, USA, 2001. ACM Press. 52

[44] Thomas K. Landauer and Susan T. Dumais. A solution to plato's problem: The latent semantic analysis theory of acquisition, induction, and representation of knowledge. *Psychological Review*, 104(2):211–240, April 1997. 7, 8, 56, 57

[45] J. R. Landis and G. G. Koch. The measurement of observer agreement for categorical data. *Biometrics*, 33:159–174, 1977. 37, 41

[46] Victor Lavrenko and W. Bruce Croft. Relevance based language models. In *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 120–127, New York, NY, USA, 2001. ACM Press.

[47] Claudia Leacock and Martin Chodorow. Combining local context and wordnet similarity for word sense identification. *An Electronic Lexical Database*, pages 265–283, 1998. 22

[48] Yoong Keok Lee and Hwee Tou Ng. An empirical evaluation of knowledge sources and learning algorithms for word sense disambiguation. In *EMNLP '02: Proceedings of the ACL-02 conference on Empirical methods in natural language processing*, pages 41–48, Morristown, NJ, USA, 2002. Association for Computational Linguistics.

[49] Michael Lesk. Automatic sense disambiguation using machine readable dictionaries: how to tell a pine cone from an ice cream cone. In *SIGDOC '86: Proceedings of the 5th annual international conference on Systems documentation*, pages 24–26, New York, NY, USA, 1986. ACM Press.

[50] Dekang Lin. An information-theoretic definition of similarity. In *ICML '98: Proceedings of the Fifteenth International Conference on Machine Learning*, pages 296–304, San Francisco, CA, USA, 1998. Morgan Kaufmann Publishers Inc.

[51] Jimmy Lin and Dina Demner-Fushman. Methods for automatically evaluating answers to complex questions. *Information Retrieval*, 9(5):565–587, 2006. 44

[52] Jimmy Lin and Dina Demner-Fushman. The role of information retrieval in answering complex questions. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 99–106, Seattle, Washington, 2006. 27, 45, 47

[53] Jimmy Lin and Dina Demner-Fushman. Will pyramids built of nuggets topple over? In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 383–390, Morristown, NJ, USA, 2006. Association for Computational Linguistics.

[54] Jimmy Lin and Pengyi Zhang. Deconstructing nuggets: The stability and reliability of complex question answering evaluation. In *Proceedings of the 30th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 327–334, Amsterdam, the Netherlands, 2007.

[55] Thomas R. Lynam, Chris Buckley, Charles L. A. Clarke, and Gordon V. Cormack. A multi-system analysis of document and term selection for blind feedback. In *CIKM '04: Proceedings of the thirteenth ACM international conference on Information and knowledge management*, pages 261–269, New York, NY, USA, 2004. ACM Press.

[56] Ian MacKinnon and Olga Vechtomova. Improving complex interactive question answering with wikipedia anchor text. In C. MacDonald et al., editor, *Proceedings of the 30th European Conference on Information Retrieval (ECIR 2008)*, pages 438–445, Glasgow, Scotland, UK, 2008. 2, 7

[57] Christopher D. Manning and Hinrich Schütze. *Foundations of Statistical Natural Language Processing*. The MIT Press, Cambridge, Massachusetts, 1999. 9

[58] K. Markert and M. Nissim. Towards a corpus annotated for metonymies: the case of location names. In *Proceedings of the 3rd International Conference on Language Resources and Evaluation*, Las Palmas, Canary Islands, May 2002. 12, 36

[59] Katja Markert and Malvina Nissim. Metonymy resolution as a classification task. In *EMNLP '02: Proceedings of the ACL-02 conference on Empirical methods in natural language processing*, pages 204–213, Morristown, NJ, USA, 2002. Association for Computational Linguistics.

[60] Gregory Marton and Alexey Radul. Nuggeteer: Automatic nugget-based evaluation using descriptions and judgements. In *Proceedings of NAACL/HLT*, 2006. 44, 47

[61] Oliver A. McBryan. GENVL and WWWW: Tools for Taming the Web. In O. Nierstarsz, editor, *Proceedings of the first International World Wide Web Conference*, page 15, CERN, Geneva, 1994.

[62] D. McCarthy, R. Koeling, J. Weeds, and J. Carroll. Finding predominant word senses in untagged text, 2004.

[63] Donald Metzler and W. Bruce Croft. A markov random field model for term dependencies. In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 472–479, New York, NY, USA, 2005. ACM Press.

[64] Rada Mihalcea. Bootstrapping large sense tagged corpora. In *Proceedings of the 3rd International Conference on Languages Resources and Evaluations LREC 2002*, Las Palmas, Spain, May 2002.

[65] Rada Mihalcea. Unsupervised large-vocabulary word sense disambiguation with graph-based algorithms for sequence data labeling. In *HLT '05: Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing*, pages 411–418, Morristown, NJ, USA, 2005. Association for Computational Linguistics.

[66] Rada Mihalcea. Using Wikipedia for automatic word sense disambiguation. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Proceedings of the Main Conference*, pages 196–203, Rochester, New York, April 2007. Association for Computational Linguistics.

[67] Rada Mihalcea, Paul Tarau, and Elizabeth Figa. Pagerank on semantic networks, with application to word sense disambiguation. In *COLING '04: Proceedings of the 20th international conference on Computational Linguistics*, page 1126, Morristown, NJ, USA, 2004. Association for Computational Linguistics.

[68] George A. Miller and Walter G. Charles. Contextual correlates of semantic similarity. *Language and Cognitive Processes*, 6(1):1–28, 1991. 16, 22

[69] Mandar Mitra, Christopher Buckley, Amit Singhal, and Claire Cardie. An analysis of statistical and syntactic phrases. In *Proceedings of RIAO-97, 5th*

International Conference "Recherche d'Information Assistee par Ordinateur", pages 200–214, Montreal, CA, 1997.

[70] Mandar Mitra, Amit Singhal, and Chris Buckley. Improving automatic query expansion. In *SIGIR '98: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval*, pages 206–214, New York, NY, USA, 1998. ACM Press.

[71] Jane Morris and Graeme Hirst. Lexical cohesion computed by thesaural relations as an indicator of the structure of text. *Comput. Linguist.*, 17(1):21–48, 1991.

[72] Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The pagerank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998. 25

[73] Han Woo Park. Hyperlink network analysis: A new method for the study of social structure on the web. *Connections*, 25(1):49–61, 2003.

[74] S. Patwardhan, S. Banerjee, and T. Pedersen. Using measures of semantic relatedness for word sense disambiguation. In *Proceedings of the Fourth International Conference on Intelligent Text Processing and Computational Linguistics*, pages 241–257, Mexico City, Mexico, February 2003.

[75] Simone Paolo Ponzetto and Michael Strube. Deriving a large scale taxonomy from wikipedia. In *Proceedings of the 22nd National Conference on Artificial Intelligence (AAAI-07)*, Vancouver, B.C., July 2007.

[76] Reid Priedhorsky, Jilin Chen, Shyong (Tony) K. Lam, Katherine Panciera, Loren Terveen, and John Riedl. Creating, destroying, and restoring value in wikipedia. In *GROUP '07: Proceedings of the 2007 international ACM conference on Supporting group work*, pages 259–268, New York, NY, USA, 2007. ACM. 21

[77] Philip Resnik. Using information content to evaluate semantic similarity in a taxonomy. In *IJCAI*, pages 448–453, 1995. 16

[78] Stephen Robertson. Understanding inverse document frequency: on theoretical arguments for idf. *Journal of Documentation*, 60:503 – 520, 2004.

[79] Stephen E. Robertson, Steve Walker, Micheline Hancock-Beaulieu, Aarron Gull, and Marianna Lau. Okapi at trec-4. In *Fourth Text REtrieval Conference (TREC-4)*, pages 73–86, 1996. 27

[80] Herbert Rubenstein and John B. Goodenough. Contextual correlates of synonymy. volume 8, pages 627–633, New York, NY, USA, 1965. ACM. 22

[81] Maria Ruiz-Casado, Enrique Alfonseca, and Pablo Castells. Automatic assignment of wikipedia encyclopedic entries to wordnet synsets. In *AWIC*, pages 380–386, 2005.

[82] Gerard Salton and Chris Buckley. Improving retrieval performance by relevance feedback. pages 355–364, 1997.

[83] Hinrich Schütze and Jan O. Pedersen. A cooccurrence-based thesaurus and two applications to information retrieval. *Inf. Process. Manage.*, 33(3):307–318, 1997.

[84] Fabrizio Sebastiani. Machine learning in automated text categorization. *ACM Computing Surveys*, 34(1):1–47, 2002.

[85] Ravi Sinha and Rada Mihalcea. Unsupervised graph-basedword sense disambiguation using measures of word semantic similarity. volume 0, pages 363–369, Los Alamitos, CA, USA, 2007. IEEE Computer Society.

[86] Fei Song and W. Bruce Croft. A general language model for information retrieval. In *CIKM '99: Proceedings of the eighth international conference on Information and knowledge management*, pages 316–321, New York, NY, USA, 1999. ACM Press.

[87] David Stallard. Two kinds of metonymy. In *Meeting of the Association for Computational Linguistics*, pages 87–94, 1993.

[88] Michael Strube and Simone P. Ponzetto. Wikirelate! computing semantic relatedness using wikipedia. In *Proceedings of the Twenty-First National Conference on Artificial Intelligence*, pages 1419–1424, Boston, Mass., July 2006. 2, 21, 24

[89] Renxu Sun, Chai-Huat Ong, and Tat-Seng Chua. Mining dependency relations for query expansion in passage retrieval. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and*

development in information retrieval, pages 382–389, New York, NY, USA, 2006. ACM Press. Modern outline for how query expansion can be done, first article looked at.

[90] Egidio Terra and C. L. A. Clarke. Frequency estimates for statistical word similarity measures. In *NAACL '03: Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology*, pages 165–172, Morristown, NJ, USA, 2003. Association for Computational Linguistics. 14

[91] P. Turney, M. L. Littman, J. Bigham, and V. Shnayder. Combining independent modules to solve multiple-choice synonym and analogy problems. In *Proceedings of International Conference on Recent Advances in Natural Language Processing (RANLP-03)*, pages 482–489, September 2003. 18, 57

[92] Peter D. Turney. Mining the web for synonyms: Pmi-ir versus lsa on toefl. In *EMCL '01: Proceedings of the 12th European Conference on Machine Learning*, pages 491–502, London, UK, 2001. Springer-Verlag. 12, 19

[93] Peter D. Turney. Similarity of semantic relations. *Computational Linguistics*, 32(3):379–416, 2006.

[94] Olga Vechtomova University. Experiments for hard and enterprise tracks.

[95] O. Vechtomova and M. Karamuftuoglu. Query expansion with terms selected using lexical cohesion analysis of documents. *Information Processing and Management*.

[96] O. Vechtomova and M. Karamuftuoglu. Use of noun phrases in interactive search refinement. In *n Proceedings of MEMURA 2004 workshop (Methodologies and Evaluation of Multiword Units in Real-world Applications), Language Resources and Evaluation Conference (LREC)*, 2004.

[97] Olga Vechtomova. The role of multi-word units in interactive information retrieval. In *Proceedings of the 27th European Conference on Information Retrieval*, pages 403–420, Santiago de Compostela, Spain, 2005.

[98] Olga Vechtomova. Noun phrases in interactive query expansion and document ranking. *Inf. Retr.*, 9(4):399–420, 2006.

[99] Olga Vechtomova and Murat Karamuftuoglu. Elicitation and use of relevance feedback information. *Inf. Process. Manage.*, 42(1):191–206, 2006.

[100] Olga Vechtomova and Murat Karamuftuoglu. Identifying relationships between entities in text for complex interactive question answering task. In *TREC*, 2006. 26, 28

[101] Olga Vechtomova and Murat Karamuftuoglu. Identifying relationships between entities in text for complex interactive question answering task. In *Proceedings of the 15th Text REtrieval Conference*, 2006.

[102] Olga Vechtomova, Murat Karamuftuoglu, and Stephen E. Robertson. On document relevance and lexical cohesion between query terms. *Inf. Process. Manage.*, 42(5):1230–1247, 2006.

[103] J. Voss. Measuring wikipedia. In *In Proc. of the 10th International Conference of the International Society for Scientometrics and Informatics*, pages 221–231, July 2005. 21

[104] B. J. Wielinga, A. Th. Schreiber, J. Wielemaker, and J. A. C. Sandberg. From thesaurus to ontology. In *K-CAP '01: Proceedings of the 1st international conference on Knowledge capture*, pages 194–201, New York, NY, USA, 2001. ACM.

[105] Jinxi Xu and W. Bruce Croft. Query expansion using local and global document analysis. In *SIGIR '96: Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 4–11, New York, NY, USA, 1996. ACM Press.

[106] Justin Zobel and Alistair Moffat. Exploring the similarity space. *SIGIR Forum*, 32(1):18–34, 1998.